



Guida per gli sviluppatori

AWS Data Pipeline



Versione API 2012-10-29

Copyright © 2023 Amazon Web Services, Inc. and/or its affiliates. All rights reserved.

AWS Data Pipeline: Guida per gli sviluppatori

Copyright © 2023 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

Cos'è AWS Data Pipeline?	1
Migrazione dei carichi di lavoro da AWS Data Pipeline	2
Migrazione dei carichi di lavoro verso AWS Glue	3
Migrazione dei carichi di lavoro a AWS Step Functions	3
Migrazione dei carichi di lavoro su Amazon MWAA	5
Mappatura dei concetti	6
Esempi	7
Servizi correlati	8
Accesso a AWS Data Pipeline	9
Prezzi	9
Tipi di istanza supportati per attività di lavoro delle pipeline	10
Istanze Amazon EC2 predefinite per regione AWS	10
Istanze Amazon EC2 aggiuntive supportate	12
Istanze Amazon EC2 supportate per i cluster Amazon EMR	12
Concetti correlati a AWS Data Pipeline	14
Definizione della pipeline	14
Componenti, istanze e tentativi della pipeline	15
Runner delle attività	17
Nodi di dati	18
Database	19
Attività	19
Precondizioni	20
Precondizioni gestite dal sistema	21
Precondizioni gestite dall'utente	21
Risorse	21
Limiti delle risorse	22
Piattaforme supportate	22
Istanze Spot Amazon EC2 con cluster Amazon EMR e AWS Data Pipeline	23
Operazioni	24
Monitoraggio proattivo delle pipeline	24
Configurazione	25
Registrazione ad AWS	25
Registrarsi per creare un Account AWS	25
Creazione di un utente amministratore	26

Crea ruoli IAM per AWS Data Pipeline risorse e pipeline	27
Consenti ai responsabili IAM (utenti e gruppi) di eseguire le azioni necessarie	27
Concessione dell'accesso programmatico	29
Nozioni di base su AWS Data Pipeline	31
Per creare la pipeline	32
Monitorare la pipeline in esecuzione	33
Visualizzazione dell'output	34
Per eliminare la pipeline	34
Uso delle pipeline	35
Creare una pipeline	35
Crea una pipeline dai modelli di Data Pipeline utilizzando la CLI	36
Visualizzazione delle pipeline	55
Interpretazione dei codici sullo stato della pipeline	55
Interpretazione dello stato di pipeline e componenti	57
Visualizzazione delle definizioni di pipeline	59
Visualizzazione dei dettagli dell'istanza della pipeline	60
Visualizza log pipeline	60
Modifica della pipeline	62
Restrizioni	62
Modifica di una pipeline tramite AWS CLI	63
Clonazione della pipeline	64
Assegnazione di tag alla pipeline	65
Disattivazione pipeline	65
Disattivare la pipeline utilizzando la AWS CLI	66
Eliminazione della pipeline	66
Dati e tabelle in gestione temporanea con attività	67
Gestione dei dati con ShellCommandActivity	68
La gestione temporanea della tabella con Hive e i nodi di dati supportati da tale gestione	70
La gestione temporanea della tabella con Hive e i nodi di dati non supportati da tale gestione	71
Utilizzo delle risorse in più regioni	72
Guasti di una delle dipendenze e riesecuzioni	75
Attività	75
Nodi di dati e condizioni preliminari	76
Risorse	76
Riesecuzione di oggetti in cascata con errore	76

Guasto in cascata e riempimenti	76
Sintassi del file di definizione della pipeline	77
Struttura dei file	77
Campi della pipeline	78
Campi definiti dall'utente	79
Lavorare con l'API	80
Installazione del kit SDK AWS	80
Effettuare una richiesta HTTP a AWS Data Pipeline	81
Sicurezza	86
Protezione dei dati	87
Identity and Access Management	88
Policy IAM per AWS Data Pipeline	89
Esempi di policy per AWS Data Pipeline	94
Ruoli IAM	97
Registrazione e monitoraggio	105
AWS Data Pipeline Informazioni in CloudTrail	105
Comprensione delle voci dei file di log di AWS Data Pipeline	106
Risposta agli eventi imprevisti	107
Convalida della conformità	108
Resilienza	108
Sicurezza dell'infrastruttura	108
Analisi della configurazione e delle vulnerabilità in AWS Data Pipeline	109
Tutorial	110
Elabora i dati utilizzando Amazon EMR con Hadoop Streaming	110
Prima di iniziare	111
Utilizzo della CLI	111
Copiare dati CSV da Amazon S3 ad Amazon S3	115
Prima di iniziare	116
Utilizzo della CLI	117
Esporta dati MySQL su Amazon S3	124
Prima di iniziare	124
Utilizzo della CLI	126
Copiare dati su Amazon Redshift	135
Prima di iniziare: configura le opzioni COPY	135
Prima di iniziare: Configura pipeline, sicurezza e cluster	136
Utilizzo della CLI	138

Funzioni ed espressioni della pipeline	148
Tipi di dati di esempio	148
DateTime	148
Numerico	148
Riferimenti agli oggetti	148
Periodo	149
Stringa	149
Espressioni	149
Riferimento a campi e oggetti	150
Espressioni nidificate	151
Elenchi	151
Espressione del nodo	152
Valutazione delle espressioni	153
Funzioni matematiche	153
Funzioni stringa	154
Funzioni di data e ora	155
Caratteri speciali	163
Riferimento all'oggetto pipeline	164
Nodi di dati	165
DynamoDB DataNode	166
MySQLDataNode	173
RedshiftDataNode	181
S3 DataNode	189
SqlDataNode	198
Attività	206
CopyActivity	207
EmrActivity	215
HadoopActivity	225
HiveActivity	237
HiveCopyActivity	247
PigActivity	257
RedshiftCopyActivity	272
ShellCommandActivity	287
SqlActivity	297
Risorse	306
Ec2Resource	306

EmrCluster	318
HttpProxy	350
Precondizioni	353
DynamoDB DataExists	354
DynamoDB TableExists	358
Exists	362
S3 KeyExists	367
S3 PrefixNotEmpty	372
ShellCommandPrecondition	376
Database	382
JdbcDatabase	382
RdsDatabase	384
RedshiftDatabase	386
Formati dei dati	389
Formato dei dati CSV	389
Formato di dati personalizzato	391
DynamoDB DataFormat	393
DynamoDB ExportDataFormat	396
RegEx Formato dei dati	398
Formato dei dati TSV	400
Azioni	402
SnsAlarm	402
Interruzione	404
Pianificazione	406
Esempi	407
Sintassi	411
Utilità	413
ShellScriptConfig	414
EmrConfiguration	415
Proprietà	420
Utilizzo di Task Runner	424
Task RunnerAWS Data Pipeline su risorse gestite	424
Esecuzione di lavori su risorse esistenti utilizzando Task Runner	426
Installazione di Task Runner	427
(Facoltativo) Concessione dell'accesso a Task Runner ad Amazon RDS	428
Avvio di Task Runner	430

Verifica della registrazione di Task Runner	431
Thread e prerequisiti di Task Runner	431
Opzioni di configurazione di Task Runner	431
Utilizzo di Task Runner con un proxy	434
Task Runner e AMI personalizzate	434
Risoluzione dei problemi	436
Individuazione di errori nelle pipeline	436
Identificazione del cluster Amazon EMR che serve la tua pipeline	437
Interpretazione dei dettagli sullo stato della pipeline	438
Individuazione dei log di errore	440
Log della pipeline	440
Step log di Hadoop Job e Amazon EMR	440
Risoluzione dei problemi più comuni	441
Pipeline bloccata in stato Pending	441
Componente della pipeline bloccato nello stato Waiting for Runner	442
Componente della pipeline bloccato nello stato WAITING_ON_DEPENDENCIES	442
L'esecuzione non inizia quando è stata programmata	443
Componenti della pipeline eseguiti in ordine errato	444
Il cluster EMR ha esito negativo con l'errore: "The security token included in the request is invalid" ("Il token di sicurezza incluso nella richiesta non è valido")	444
Autorizzazioni insufficienti per accedere alle risorse	444
Codice di stato: 400 Codice di errore: PipelineNotFoundException	445
La creazione di una pipeline provoca un errore relativo al Security Token	445
Impossibile visualizzare i dettagli della pipeline nella console	445
Errore in remote runner Codice stato: 404, AWS Service: Amazon S3	445
Accesso negato - Non autorizzato per eseguire la funzione datapipeline:	445
Le AMI Amazon EMR meno recenti possono creare dati falsi per file CSV di grandi dimensioni	446
Aumento dei limiti AWS Data Pipeline	447
Limiti	448
Limiti dell'account	448
Limiti chiamata del servizio Web	449
Considerazioni su dimensionamento	451
Risorse di AWS Data Pipeline	452
Cronologia dei documenti	454
.....	cdlx

Cos'è AWS Data Pipeline?

Note

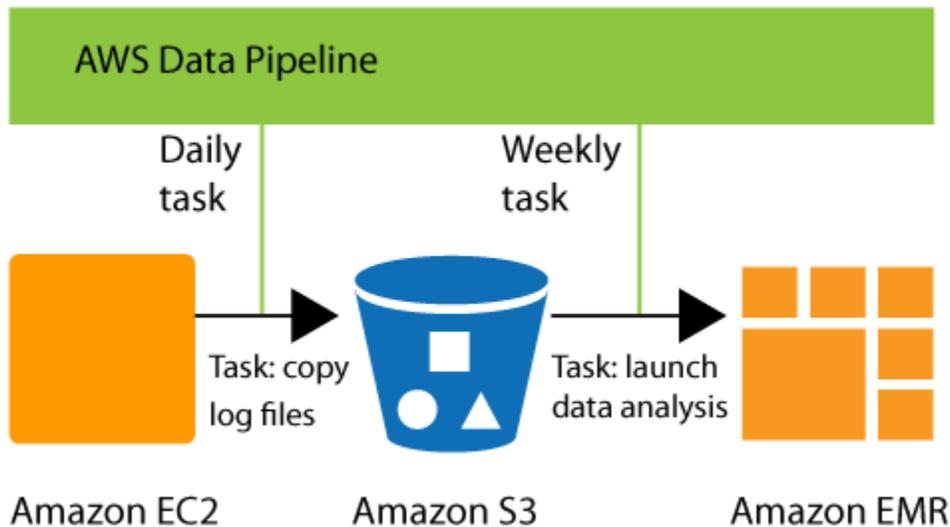
AWS Data Pipeline il servizio è in modalità di manutenzione e non sono previste nuove funzionalità o espansioni regionali. Per ulteriori informazioni e per scoprire come migrare i carichi di lavoro esistenti, consulta [Migrazione dei carichi di lavoro da AWS Data Pipeline](#)

AWS Data Pipeline è un servizio Web che automatizza il movimento e la trasformazione dei dati. Con AWS Data Pipeline, si possono definire i flussi di lavoro basati sui dati, in modo che le attività possano essere dipendenti dal completamento delle attività precedenti. È possibile definire i parametri delle trasformazioni dei dati e AWS Data Pipeline applica la logica impostata.

I seguenti componenti di AWS Data Pipeline collaborano per gestire i dati:

- Una pipeline definition (definizione di pipeline) specifica la logica di business della gestione dei dati. Per ulteriori informazioni, consulta [Sintassi del file di definizione della pipeline](#).
- Una pipeline pianifica ed esegue le attività creando istanze Amazon EC2 per eseguire le attività lavorative definite. È possibile caricare la definizione di pipeline nella e quindi attivarla. È possibile modificare la definizione di pipeline e attivare di nuovo la pipeline affinché abbia effetto. Puoi disattivare la pipeline, modificare un'origine dati e quindi attivare la pipeline di nuovo. Quando la pipeline non è più necessaria, è possibile eliminarla.
- Task Runner effettua un sondaggio per le attività e quindi le esegue. Ad esempio, Task Runner potrebbe copiare i file di registro su Amazon S3 e avviare cluster Amazon EMR. Task Runner viene installato ed eseguito automaticamente sulle risorse create dalle definizioni della pipeline. È possibile scrivere un'applicazione task runner personalizzata oppure utilizzare l'applicazione Task Runner fornita da AWS Data Pipeline. Per ulteriori informazioni, consulta [Runner delle attività](#).

Ad esempio, puoi AWS Data Pipeline archiviare i log del tuo server web su Amazon Simple Storage Service (Amazon S3) ogni giorno e quindi eseguire un cluster Amazon EMR (Amazon EMR) settimanale su tali registri per generare report sul traffico. AWS Data Pipeline pianifica le attività giornaliere per copiare i dati e l'attività settimanale per avviare il cluster Amazon EMR. AWS Data Pipeline garantisce inoltre che Amazon EMR attenda il caricamento dei dati dell'ultimo giorno su Amazon S3 prima di iniziare l'analisi, anche in caso di ritardo imprevisto nel caricamento dei log.



Indice

- [Migrazione dei carichi di lavoro da AWS Data Pipeline](#)
- [Servizi correlati](#)
- [Accesso a AWS Data Pipeline](#)
- [Prezzi](#)
- [Tipi di istanza supportati per attività di lavoro delle pipeline](#)

Migrazione dei carichi di lavoro da AWS Data Pipeline

AWS ha lanciato il AWS Data Pipeline servizio nel 2012. A quel tempo, i clienti cercavano un servizio che li aiutasse a spostare in modo affidabile i dati tra diverse fonti di dati utilizzando una varietà di opzioni di elaborazione. Ora, ci sono altri servizi che offrono ai clienti un'esperienza migliore. Ad esempio, puoi utilizzare per eseguire e AWS Glue orchestrare applicazioni Apache Spark, AWS Step Functions per aiutare a orchestrare i componenti del AWS servizio o Amazon Managed Workflows for Apache Airflow (Amazon MWAA) per aiutare a gestire l'orchestrazione del flusso di lavoro per Apache Airflow.

Questo argomento spiega come eseguire la migrazione AWS Data Pipeline da opzioni alternative. L'opzione scelta dipende dal carico di lavoro corrente. AWS Data Pipeline Puoi migrare i casi d'uso tipici AWS Data Pipeline su AWS Step Functions o Amazon MWAA. AWS Glue

Migrazione dei carichi di lavoro verso AWS Glue

[AWS Glue](#) è un servizio di integrazione dati serverless che semplifica agli utenti analitici il rilevamento, la preparazione, lo spostamento e l'integrazione di dati da più origini. Include strumenti per la creazione, l'esecuzione di processi e l'orchestrazione dei flussi di lavoro. Con AWS Glue puoi rilevare e collegarti a oltre 70 diverse origini di dati e gestire i tuoi dati in un catalogo dati centralizzato. Puoi creare, eseguire e monitorare visivamente pipeline di estrazione, trasformazione e caricamento (ETL) per caricare dati nei data lake. Inoltre, puoi eseguire ricerche e query immediatamente nei dati catalogati utilizzando Amazon Athena, Amazon EMR e Amazon Redshift Spectrum.

Ti consigliamo di migrare il AWS Data Pipeline carico di lavoro a AWS Glue quando:

- Stai cercando un servizio di integrazione dei dati senza server che supporti varie fonti di dati, interfacce di creazione tra cui editor visivi e notebook e funzionalità avanzate di gestione dei dati come la qualità dei dati e il rilevamento di dati sensibili.
- Il carico di lavoro può essere migrato verso AWS Glue flussi di lavoro, job (in Python o Apache Spark) e crawler (ad esempio, la pipeline esistente è basata su Apache Spark).
- È necessaria un'unica piattaforma in grado di gestire tutti gli aspetti della pipeline di dati, tra cui acquisizione, elaborazione, trasferimento, test di integrità e controlli di qualità.
- La tua pipeline esistente è stata creata da un modello predefinito sulla AWS Data Pipeline console, ad esempio l'esportazione di una tabella DynamoDB in Amazon S3, e stai cercando lo stesso modello per lo stesso scopo.
- Il carico di lavoro non dipende da una specifica applicazione dell'ecosistema Hadoop come Apache Hive.
- Il carico di lavoro non richiede l'orchestrazione di server locali.

AWS addebita una tariffa oraria, fatturata al secondo, per i crawler (rilevamento dei dati) e i job ETL (elaborazione e caricamento dei dati). AWS Glue Studio è un motore di orchestrazione integrato per AWS Glue le risorse e viene offerto senza costi aggiuntivi. Scopri di più sui prezzi nella sezione [AWS GluePrezzi](#).

Migrazione dei carichi di lavoro a AWS Step Functions

[AWS Step Functions](#) è un servizio di orchestrazione senza server che consente di creare flussi di lavoro per le applicazioni aziendali critiche. Con Step Functions utilizzi un editor visivo per creare flussi di lavoro e integrarli direttamente con oltre 11.000 azioni per oltre 250 AWS servizi,

come AWS Lambda, Amazon EMR, DynamoDB e altri. È possibile utilizzare Step Functions per orchestrare le pipeline di elaborazione dei dati, gestire gli errori e gestire i limiti di limitazione dei servizi sottostanti. AWS Puoi creare flussi di lavoro che elaborano e pubblicano modelli di machine learning, orchestrano microservizi e AWS servizi di controllo, ad esempio per creare flussi di lavoro di estrazione, trasformazione e caricamento (ETL). AWS Glue Puoi anche creare flussi di lavoro automatizzati e di lunga durata per applicazioni che richiedono l'interazione umana.

Analogamente a AWS Data Pipeline, AWS Step Functions è un servizio completamente gestito fornito da AWS. Non sarà necessario gestire l'infrastruttura, i patch worker, gestire gli aggiornamenti delle versioni del sistema operativo o simili.

Ti consigliamo di migrare il AWS Data Pipeline carico di lavoro a AWS Step Functions quando:

- Stai cercando un servizio di orchestrazione del flusso di lavoro senza server e ad alta disponibilità.
- Stai cercando una soluzione conveniente che preveda la granularità dell'esecuzione di una singola attività.
- I tuoi carichi di lavoro orchestrano le attività per diversi altri AWS servizi, come Amazon EMR, Lambda o DynamoDB. AWS Glue
- Stai cercando una soluzione low-code dotata di un drag-and-drop visual designer per la creazione di flussi di lavoro e che non richieda l'apprendimento di nuovi concetti di programmazione.
- Stai cercando un servizio che fornisca integrazioni con oltre 250 altri AWS servizi che coprano oltre 11.000 azioni out-of-the-box, oltre a consentire integrazioni con attività e servizi non AWS servizi personalizzati.

AWS Data Pipeline Sia Step Functions che Step utilizzano il formato JSON per definire i flussi di lavoro. Ciò consente di archiviare i flussi di lavoro nel controllo del codice sorgente, gestire le versioni, controllare l'accesso e automatizzare con CI/CD. Step Functions utilizza una sintassi chiamata Amazon State Language che è completamente basata su JSON e consente una transizione senza interruzioni tra le rappresentazioni testuali e visive del flusso di lavoro.

Con Step Functions, puoi scegliere la stessa versione di Amazon EMR in AWS Data Pipeline cui stai attualmente utilizzando.

Per la migrazione delle attività su risorse AWS Data Pipeline gestite, puoi utilizzare [l'integrazione dei servizi AWS SDK](#) su Step Functions per automatizzare la fornitura e la pulizia delle risorse.

[Per la migrazione delle attività su server locali, istanze EC2 gestite dagli utenti o un cluster EMR gestito dagli utenti, puoi installare un agente SSM sull'istanza.](#) È possibile avviare il comando tramite

[AWS Systems Manager Run Command](#) da Step Functions. Puoi anche avviare la macchina a stati dalla pianificazione definita in [Amazon EventBridge](#).

AWS Step Functions dispone di due tipi di flussi di lavoro: flussi di lavoro standard e flussi di lavoro rapidi. Per i flussi di lavoro standard, i costi vengono calcolati in base al numero di transizioni di stato necessarie per eseguire l'applicazione. Per Express Workflow, ti vengono addebitati in base al numero di richieste per il flusso di lavoro e alla sua durata. Scopri di più sui prezzi nella sezione Prezzi di [AWS Step Functions](#).

Migrazione dei carichi di lavoro su Amazon MWAA

[Amazon MWAA](#) (Managed Workflows for Apache Airflow) è un servizio di orchestrazione gestito per [Apache Airflow](#) che semplifica la configurazione e la gestione di pipeline di dati end-to-end nel cloud su larga scala. Apache Airflow è uno strumento open source utilizzato per creare, pianificare e monitorare in modo programmatico sequenze di processi e attività denominate «flussi di lavoro». Con Amazon MWAA, puoi utilizzare i linguaggi di programmazione Airflow e Python per creare flussi di lavoro senza dover gestire l'infrastruttura sottostante per scalabilità, disponibilità e sicurezza. Amazon MWAA ridimensiona automaticamente la capacità di esecuzione del flusso di lavoro per soddisfare le tue esigenze ed è integrato con i servizi AWS di sicurezza per aiutarti a fornire un accesso rapido e sicuro ai tuoi dati.

Analogamente a AWS Data Pipeline, Amazon MWAA è un servizio completamente gestito fornito da AWS. Sebbene sia necessario apprendere diversi nuovi concetti specifici di questi servizi, non è necessario gestire l'infrastruttura, i patch worker, gestire gli aggiornamenti delle versioni del sistema operativo o simili.

Ti consigliamo di migrare i tuoi AWS Data Pipeline carichi di lavoro su Amazon MWAA quando:

- Stai cercando un servizio gestito e altamente disponibile per orchestrare i flussi di lavoro scritti in Python.
- Vuoi passare a una tecnologia open source completamente gestita e ampiamente adottata, Apache Airflow, per la massima portabilità.
- È necessaria un'unica piattaforma in grado di gestire tutti gli aspetti della pipeline di dati, tra cui acquisizione, elaborazione, trasferimento, test di integrità e controlli di qualità.
- Stai cercando un servizio progettato per l'orchestrazione delle pipeline di dati con funzionalità come un'interfaccia utente avanzata per l'osservabilità, riavvii per flussi di lavoro non riusciti, backfill e nuovi tentativi per le attività.

- Stai cercando un servizio che include più di 800 operatori e sensori predefiniti, che copra AWS anche i AWS servizi non.

I flussi di lavoro Amazon MWAA sono definiti come grafici aciclici diretti (DAG) utilizzando Python, quindi puoi anche trattarli come codice sorgente. Il framework Python estensibile di Airflow consente di creare flussi di lavoro che si connettono praticamente con qualsiasi tecnologia. È dotato di una ricca interfaccia utente per la visualizzazione e il monitoraggio dei flussi di lavoro e può essere facilmente integrato con i sistemi di controllo delle versioni per automatizzare il processo CI/CD.

Con Amazon MWAA, puoi scegliere la stessa versione di Amazon EMR in cui stai attualmente utilizzando. AWS Data Pipeline

AWS addebita la durata di funzionamento dell'ambiente Airflow, oltre a qualsiasi ulteriore ridimensionamento automatico per aumentare la capacità del personale o del server Web. Scopri di più sui prezzi in [Amazon Managed Workflows per Apache Airflow](#).

Mappatura dei concetti

La tabella seguente contiene la mappatura dei concetti principali utilizzati dai servizi. Aiuterà le persone che hanno familiarità con Data Pipeline a comprendere Step Functions e la terminologia MWAA.

Data Pipeline	Aderenza	Step Functions	Amazon MWAA
Pipeline	Flussi di lavoro	Flussi di lavoro	Grafici acrilici diretti
Definizione della pipeline JSON	Definizione del flusso di lavoro o blueprint basati su Python	Amazon State Language JSON	Basato su Python
Attività	Jobs	Stati e compiti	Compiti (operatori e sensori)
Istanze	Esecuzione del lavoro	Esecuzioni	DAG corre
Tentativi	Tentativi di nuovo tentativo	Catchers e retriers	Tentativi

Data Pipeline	Aderenza	Step Functions	Amazon MWAA
Pianificazione della pipeline	Pianifica i trigger	EventBridgeAttività di pianificazione	Cron, orari, data-aware e
Espressioni e funzioni della pipeline	Libreria Blueprint	Step Functions, funzioni intrinseche e Lambda AWS	Framework Python estensibile

Esempi

Le sezioni seguenti elencano esempi pubblici a cui è possibile fare riferimento per la migrazione AWS Data Pipeline da servizi individuali. Puoi citarli come esempi e creare la tua pipeline sui singoli servizi aggiornandola e testandola in base al tuo caso d'uso.

Esempi di AWS Glue

L'elenco seguente contiene esempi di implementazioni per i casi AWS Data Pipeline d'uso più comuni con AWS Glue

- [Esecuzione di lavori Spark](#)
- [Copiare dati da JDBC ad Amazon S3 \(incluso Amazon Redshift\)](#)
- [Copiare dati da Amazon S3 a JDBC \(incluso Amazon Redshift\)](#)
- [Copiare dati da Amazon S3 a DynamoDB](#)
- [Trasferimento di dati da e verso Amazon Redshift](#)
- [Accesso interregionale alle tabelle DynamoDB tra più account](#)

AWS Esempi di Step Functions

L'elenco seguente contiene esempi di implementazioni per i AWS Data Pipeline casi d'uso più comuni con AWS Step Functions.

- [Gestione di un processo Amazon EMR](#)
- [Esecuzione di un processo di elaborazione dei dati su Amazon EMR Serverless](#)
- [Esecuzione di processi Hive/Pig/Hadoop](#)

- [Interrogazione di set di dati di grandi dimensioni](#) (Amazon Athena, Amazon S3) AWS Glue
- [Esecuzione di flussi di lavoro ETL con Amazon Redshift](#)
- [Orchestratura AWS Glue dei crawler](#)

Guarda [tutorial](#) aggiuntivi e [progetti di esempio per l'utilizzo di](#) AWS Step Functions.

Esempi Amazon MWAA

L'elenco seguente contiene esempi di implementazioni per i AWS Data Pipeline casi d'uso più comuni con Amazon MWAA.

- [Esecuzione di un processo Amazon EMR](#)
- [Creazione di un plugin personalizzato per Apache Hive e Hadoop](#)
- [Copiare dati da Amazon S3 a Redshift](#)
- [Esecuzione di uno script Shell su un'istanza EC2 remota](#)
- [Orchestratura di flussi di lavoro ibridi \(on-premise\)](#)

Guarda [tutorial](#) aggiuntivi e [progetti](#) di esempio per l'utilizzo di Amazon MWAA.

Servizi correlati

AWS Data Pipeline lavora con i seguenti servizi per archiviare i dati.

- Amazon DynamoDB: fornisce un database NoSQL completamente gestito con prestazioni veloci a basso costo. Per ulteriori informazioni, consulta la [Amazon DynamoDB Developer Guide](#).
- Amazon RDS: fornisce un database relazionale completamente gestito scalabile fino a set di dati di grandi dimensioni. Per ulteriori informazioni, consulta la [Guida per gli sviluppatori di Amazon Relational Database Service](#).
- Amazon Redshift: fornisce un data warehouse veloce, completamente gestito e su scala di petabyte che semplifica ed economico l'analisi di una grande quantità di dati. Per ulteriori informazioni, consulta la [Amazon Redshift Database Developer Guide](#).
- Amazon S3: fornisce uno storage di oggetti sicuro, durevole e altamente scalabile. Per ulteriori informazioni, consulta la [Guida per l'utente di Amazon Simple Storage Service](#).

AWS Data Pipeline lavora con i seguenti servizi di elaborazione per trasformare i dati.

- Amazon EC2: fornisce una capacità di elaborazione ridimensionabile, letteralmente server nei data center di Amazon, che puoi utilizzare per creare e ospitare i tuoi sistemi software. Per ulteriori informazioni, consulta la [Guida per l'utente di Amazon EC2 per le istanze Linux](#).
- Amazon EMR: semplifica, veloce ed economico la distribuzione e l'elaborazione di grandi quantità di dati tra i server Amazon EC2, utilizzando un framework come Apache Hadoop o Apache Spark. Per ulteriori informazioni, consulta la [Amazon EMR Developer Guide](#).

Accesso a AWS Data Pipeline

È possibile creare, accedere e gestire le pipeline utilizzando una qualsiasi delle seguenti interfacce:

- AWS Management Console— Fornisce un'interfaccia web a cui è possibile accedere AWS Data Pipeline.
- AWS Command Line Interface (AWS CLI) — Fornisce comandi per un'ampia gamma di servizi AWS Data Pipeline, inclusi ed è supportato su Windows, macOS e Linux. Per ulteriori informazioni sull'installazione della AWS CLI consulta [AWS Command Line Interface](#). Per un elenco dei comandi per AWS Data Pipeline consulta [datapipeline](#).
- SDK AWS - Forniscono interfacce API specifiche per ogni linguaggio e si prendono carico di molti dettagli della connessione, ad esempio il calcolo delle firme e la gestione dei tentativi di richiesta e degli errori. Per ulteriori informazioni, consulta [SDK AWS](#).
- API di interrogazione: fornisce API di basso livello che puoi chiamare utilizzando richieste HTTPS. L'API di interrogazione è il modo più diretto per accedere al AWS Data Pipeline, ma richiede che la propria applicazione gestisca dettagli di basso livello, come la generazione di un hash per la firma della richiesta e la gestione degli errori. Per ulteriori informazioni, consulta la Documentazione di riferimento delle API di [AWS Data Pipeline](#).

Prezzi

I prezzi di Amazon Web Services sono calcolati in base all'uso effettivo. Per AWS Data Pipeline il costo della pipeline dipende dalla frequenza con cui è prevista l'esecuzione di attività e precondizioni e dall'ambiente in cui vengono eseguite. Per ulteriori informazioni, consultare [Prezzi di AWS Data Pipeline](#).

Se l'account AWS è inferiore a 12 mesi, hai diritto a utilizzare il piano gratuito. Il piano gratuito include tre precondizioni a bassa frequenza e cinque attività a bassa frequenza al mese senza alcun costo aggiuntivo. Per ulteriori informazioni, consulta [Piano gratuito di AWS](#).

Tipi di istanza supportati per attività di lavoro delle pipeline

Quando AWS Data Pipeline esegue una pipeline, compila i componenti della pipeline per creare un set di istanze Amazon EC2 utilizzabili. Ogni istanza contiene tutte le informazioni necessarie per l'esecuzione di un'attività specifica. Il set completo di istanze costituisce l'elenco delle attività della pipeline. AWS Data Pipeline passa le istanze ai runner delle attività per essere processate.

Le istanze EC2 sono disponibili in diverse configurazioni, note come tipi di istanze. Ogni tipo di istanza dispone di diverse capacità di CPU, input/output e storage. Oltre a specificare il tipo di istanza per un'attività, puoi scegliere diverse opzioni di acquisto. Non tutti i tipi di istanze sono disponibili in tutte le regioni AWS. Se un tipo di istanza non è disponibile, è possibile che la pipeline non riesca a effettuare il provisioning o che il provisioning si blocchi. Per informazioni sulla disponibilità delle istanze, consulta la [pagina dei prezzi di Amazon EC2](#). Aprire il link per l'opzione di acquisto di istanze e filtrare in base alla Region (Regione) per vedere se un tipo di istanza è disponibile nella regione. Per ulteriori informazioni su questi tipi di istanze, famiglie e tipi di virtualizzazione, consulta le istanze [Amazon EC2 e la matrice dei tipi di istanze AMI di Amazon Linux](#).

Nella seguente tabella vengono descritti i tipi di istanze supportati da AWS Data Pipeline. Puoi utilizzarlo AWS Data Pipeline per avviare istanze Amazon EC2 in qualsiasi regione, comprese le regioni in cui non AWS Data Pipeline è supportata. Per informazioni sulle regioni in cui è supportato AWS Data Pipeline, consulta [Regioni ed endpoint AWS](#).

Indice

- [Istanze Amazon EC2 predefinite per regione AWS](#)
- [Istanze Amazon EC2 aggiuntive supportate](#)
- [Istanze Amazon EC2 supportate per i cluster Amazon EMR](#)

Istanze Amazon EC2 predefinite per regione AWS

Se non si specifica un tipo di istanza nella definizione della pipeline, AWS Data Pipeline avvia un'istanza per impostazione predefinita.

La tabella seguente elenca le istanze Amazon EC2 AWS Data Pipeline utilizzate di default nelle regioni in cui AWS Data Pipeline è supportata.

Nome della regione	Regione	Tipo di istanza
Stati Uniti orientali (Virginia settentrionale)	us-east-1	m1.small
Stati Uniti occidentali (Oregon)	us-west-2	m1.small
Asia Pacifico (Sydney)	ap-southeast-2	m1.small
Asia Pacifico (Tokyo)	ap-northeast-1	m1.small
UE (Irlanda)	eu-west-1	m1.small

La tabella seguente elenca le istanze Amazon EC2 che vengono AWS Data Pipeline avviate per impostazione predefinita nelle regioni in cui AWS Data Pipeline non sono supportate.

Nome della regione	Regione	Tipo di istanza
Stati Uniti orientali (Ohio)	us-east-2	t2.small
Stati Uniti occidentali (California settentrionale)	us-west-1	m1.small
Asia Pacifico (Mumbai)	ap-south-1	t2.small
Asia Pacifico (Singapore)	ap-southeast-1	m1.small
Asia Pacifico (Seoul)	ap-northeast-2	t2.small
Canada (Centrale)	ca-central-1	t2.small
UE (Francoforte)	eu-central-1	t2.small
UE (Londra)	eu-west-2	t2.small
UE (Parigi)	eu-west-3	t2.small
Sud America (San Paolo)	sa-east-1	m1.small

Istanze Amazon EC2 aggiuntive supportate

Oltre alle istanze predefinite che vengono create se non si specifica un tipo di istanza nella definizione della pipeline, vengono supportate le seguenti istanze.

La tabella seguente elenca le istanze Amazon EC2 che AWS Data Pipeline supportano e possono creare, se specificato.

Classe istanza	Tipi di istanza
Uso generale	t2.nano t2.micro t2.small t2.medium t2.large
Calcolo ottimizzato	c3.large c3.xlarge c3.2xlarge c3.4xlarge c3.8xlarge c4.large c4.xlarge c4.2xlarge c4.4xlarge c4.8xlarge c5.xlarge c5.9xlarge c5.2xlarge c5.4xlarge c5.9xlarge c5.18xlarge c5d.xlarge c5d.2xlarge c5d.4xlarge c5d.9xlarge c5d.18xlarge
Memoria ottimizzata	m3.medium m3.large m3.xlarge m3.2xlarge m4.large m4.xlarge m4.2xlarge m4.4xlarge m4.10xlarge m4.16xlarge m5.xlarge m5.2xlarge m5.4xlarge m5.12xlarge m5.24xlar ge m5d.xlarge m5d.2xlarge m5d.4xlarge m5d.12xlarge m5d.24xlarge r3.large r3.xlarge r3.2xlarge r3.4xlarge r3.8xlarge r4.large r4.xlarge r4.2xlarge r4.4xlarge r4.8xlarge r4.16xlarge
Storage ottimizzato	i2.xlarge i2.2xlarge i2.4xlarge i2.8xlarge hs1.8xlarge g2.2xlarge g2.8xlarge d2.xlarge d2.2xlarge d2.4xlarge d2.8xlarge

Istanze Amazon EC2 supportate per i cluster Amazon EMR

Questa tabella elenca le istanze Amazon EC2 che AWS Data Pipeline supportano e possono creare per i cluster Amazon EMR, se specificato. Per ulteriori informazioni, consulta [Tipi di istanza supportati](#) nella Guida alla gestione di Amazon EMR.

Classe istanza	Tipi di istanza
Uso generale	m1.small m1.medium m1.large m1.xlarge m3.xlarge m3.2xlarge
Calcolo ottimizzato	c1.medium c1.xlarge c3.xlarge c3.2xlarge c3.4xlarge c3.8xlarge cc1.4xlarge cc2.8xlarge c4.large c4.xlarge c4.2xlarge c4.4xlarge c4.8xlarge c5.xlarge c5.9xlarge c5.2xlarge c5.4xlarge c5.9xlarge c5.18xlarge c5d.xlarge c5d.2xlarge c5d.4xlarge c5d.9xlarge c5d.18xlarge
Memoria ottimizzata	m2.xlarge m2.2xlarge m2.4xlarge r3.xlarge r3.2xlarge r3.4xlarge r3.8xlarge cr1.8xlarge m4.large m4.xlarge m4.2xlarge m4.4xlarge m4.10xlarge m4.16large m5.xlarge m5.2xlarge m5.4xlarge m5.12xlarge m5.24xlarge m5d.xlarge m5d.2xlarge m5d.4xlarge m5d.12xlarge m5d.24xlarge r4.large r4.xlarge r4.2xlarge r4.4xlarge r4.8xlarge r4.16xlarge
Storage ottimizzato	h1.4xlarge hs1.2xlarge hs1.4xlarge hs1.8xlarge i2.xlarge i2.2xlarge i2.4xlarge i2.8xlarge d2.xlarge d2.2xlarge d2.4xlarge d2.8xlarge
Accelerazione informatica	g2.2xlarge cg1.4xlarge

Concetti correlati a AWS Data Pipeline

Prima di iniziare, leggere i componenti e i concetti fondamentali di AWS Data Pipeline.

Indice

- [Definizione della pipeline](#)
- [Componenti, istanze e tentativi della pipeline](#)
- [Runner delle attività](#)
- [Nodi di dati](#)
- [Database](#)
- [Attività](#)
- [Precondizioni](#)
- [Risorse](#)
- [Operazioni](#)

Definizione della pipeline

Una definizione della pipeline consiste nel modo di comunicare la logica di business ad AWS Data Pipeline. Contiene le seguenti informazioni:

- Nomi, percorsi e formati delle origini dati
- Attività per la trasformazione dei dati
- La pianificazione per tali attività
- Risorse che eseguono attività e precondizioni
- Le precondizioni devono essere soddisfatte prima che le attività possano essere programmate
- Metodi per avvisarti con aggiornamenti di stato durante l'esecuzione della pipeline

Dalla definizione della pipeline, AWS Data Pipeline stabilisce le attività, le pianifica e le assegna ai runner delle attività. Se un'attività non viene completata correttamente, AWS Data Pipeline prova di nuovo l'attività in base alle istruzioni e, se necessario, la riassegna a un altro runner di attività. Se l'operazione ha esito negativo ripetutamente, è possibile configurare la pipeline per la notifica.

Ad esempio, nella definizione della pipeline, è possibile specificare che i file di registro generati dall'applicazione vengano archiviati ogni mese nel 2013 in un bucket Amazon S3. AWS Data Pipeline creerebbe quindi 12 attività, ognuna delle quali copie più di un mese di dati, indipendentemente dal fatto che il mese contenga 30, 31, 28 o 29 giorni.

Puoi creare una definizione di pipeline nei seguenti modi:

- Graficamente, utilizzando la console AWS Data Pipeline
- Testualmente, scrivendo un file in formato JSON utilizzato dall'interfaccia a riga di comando
- A livello di pianificazione, chiamando il servizio Web con uno dei kit SDK AWS oppure [l'API AWS Data Pipeline](#)

Una definizione di pipeline può contenere i seguenti tipi di componenti.

Componenti della pipeline

[Nodi di dati](#)

La posizione dei dati di input per un'attività o il percorso in cui i dati di output vengono archiviati.

[Attività](#)

Una definizione di lavoro da eseguire in base a una pianificazione utilizzando una risorsa di calcolo e, in genere, i nodi di dati di input e di output.

[Precondizioni](#)

Un'istruzione condizionale che deve essere true prima di eseguire un'operazione.

[Risorse](#)

La risorsa di calcolo che esegue il lavoro definito da una pipeline.

[Operazioni](#)

Operazione che viene attivata quando vengono soddisfatte determinate condizioni, per esempio la non riuscita di un'attività.

Per ulteriori informazioni, consulta [Sintassi del file di definizione della pipeline](#).

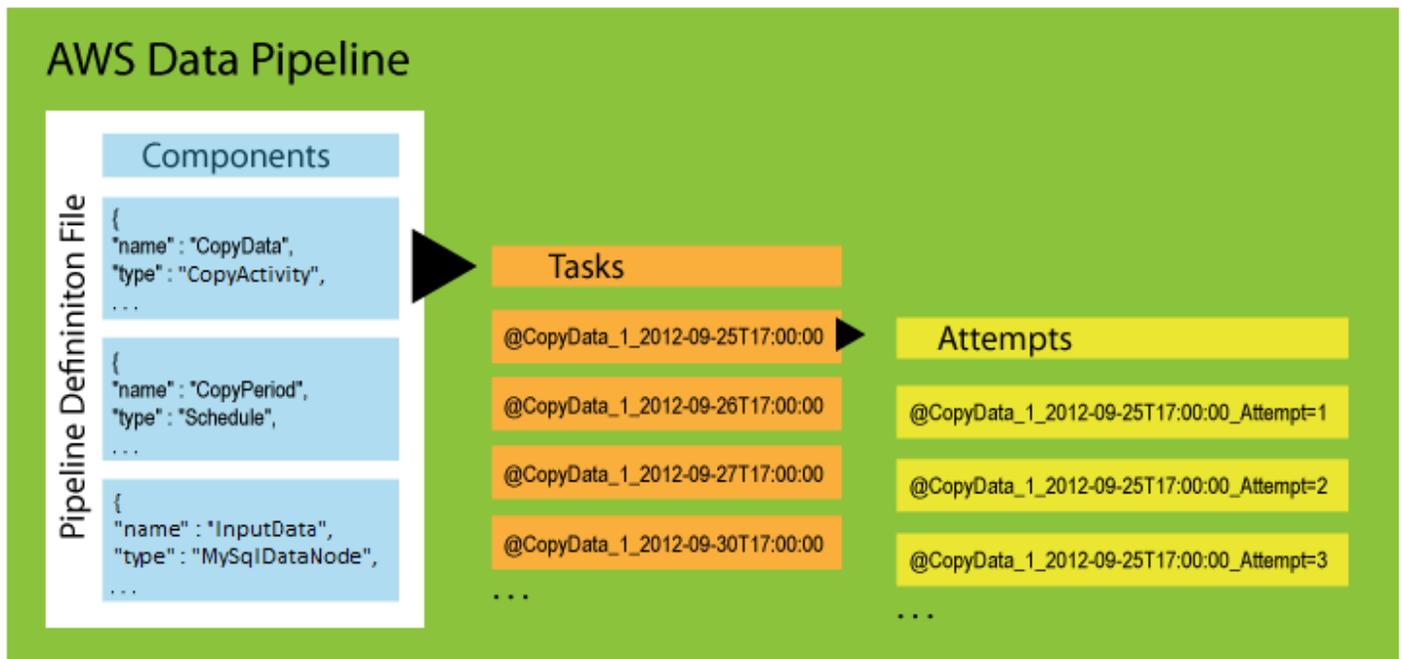
Componenti, istanze e tentativi della pipeline

Esistono tre tipi di elementi associati a una pipeline pianificata:

- **Componenti della pipeline:** i componenti della pipeline rappresentano la logica aziendale della pipeline e sono rappresentati dalle diverse sezioni della definizione di pipeline. I componenti della pipeline specificano le origini dati, le attività, la pianificazione e le precondizioni del flusso di lavoro. È possibile ereditare le proprietà da componenti padre. Relazioni tra i componenti vengono definite per riferimento. I componenti della pipeline definiscono le regole di gestione dei dati.
- **Istanze:** quando AWS Data Pipeline esegue una pipeline, compila i componenti della pipeline per creare una serie di istanze utilizzabili. Ogni istanza contiene tutte le informazioni necessarie per l'esecuzione di un'attività specifica. Il set completo di istanze costituisce l'elenco delle attività della pipeline. AWS Data Pipeline passa le istanze ai runner delle attività per essere processate.
- **Tentativi:** per fornire una solida gestione dei dati, AWS Data Pipeline riprova un'operazione non riuscita. Continua a farlo finché l'attività non raggiunge il numero massimo di tentativi consentiti. Gli Attempt Objects monitorano i diversi tentativi, i risultati e i motivi di errore, ove applicabile. Essenzialmente, è l'istanza con un contatore. AWS Data Pipeline esegue nuovi tentativi utilizzando le stesse risorse dei tentativi precedenti, come i cluster Amazon EMR e le istanze EC2.

Note

Rieseguire le attività non riuscite è un aspetto importante di una strategia di tolleranza ai guasti e le definizioni di AWS Data Pipeline forniscono condizioni e soglie per controllare i nuovi tentativi. Tuttavia, troppi tentativi possono ritardare il rilevamento di un errore irreversibile perché AWS Data Pipeline non segnala il guasto finché non ha esaurito tutti i tentativi specificati dall'utente. I nuovi tentativi aggiuntivi possono comportare costi aggiuntivi se sono in esecuzione su risorse AWS. Di conseguenza, valutare attentamente quando è opportuno superare le impostazioni predefinite di AWS Data Pipeline utilizzate per il controllo dei nuovi tentativi e delle relative impostazioni.

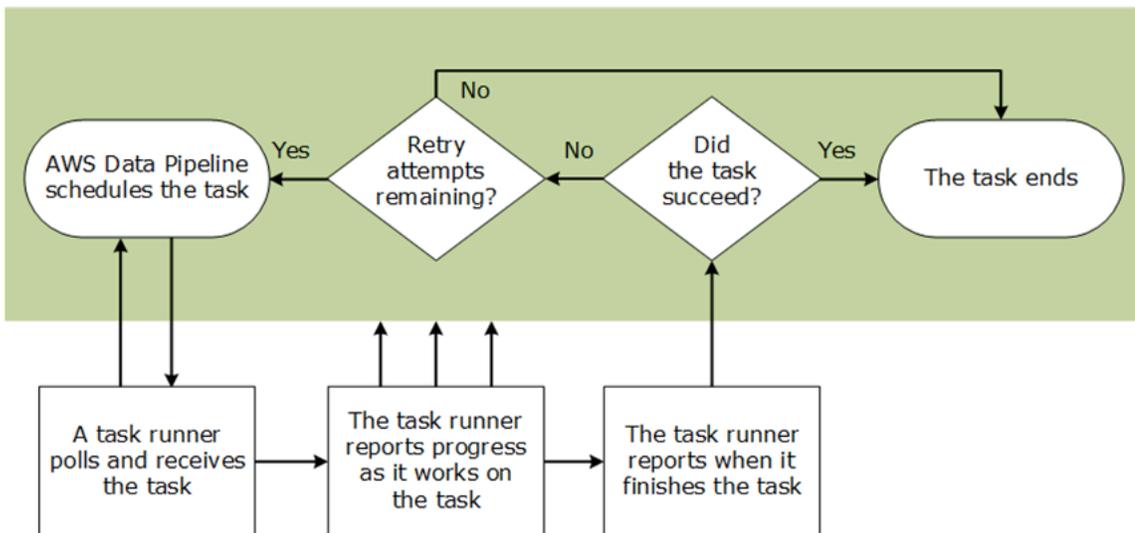


Runner delle attività

Un runner di attività è un'applicazione che esegue il polling di AWS Data Pipeline per le attività, quindi esegue tali operazioni.

Task Runner è un'implementazione predefinita di un task runner fornito da AWS Data Pipeline. Quando Task Runner è installato e configurato, esegue un sondaggio AWS Data Pipeline per le attività associate alle pipeline che hai attivato. Quando un'attività viene assegnata a Task Runner, esegue tale attività e ne riporta lo stato a AWS Data Pipeline.

Il seguente diagramma illustra come AWS Data Pipeline e un runner dell'attività interagisce per elaborare un'attività pianificata. Un'attività è un'unità discreta di lavoro che il servizio AWS Data Pipeline condivide con un runner di attività. È diversa da una pipeline, che è una definizione generale di attività e risorse che normalmente produce diverse attività.



Esistono due modi per utilizzare Task Runner per elaborare la pipeline:

- AWS Data Pipeline installa Task Runner automaticamente sulle risorse avviate e gestite dal servizio AWS Data Pipeline web.
- Task Runner viene installato su una risorsa computazionale gestita dall'utente, ad esempio un'istanza EC2 di lunga durata o un server locale.

Per ulteriori informazioni sull'utilizzo di Task Runner, consulta [Utilizzo di Task Runner](#).

Nodi di dati

In AWS Data Pipeline, un nodo di dati definisce la posizione e il tipo di dati che utilizza una pipeline di attività come input o output. AWS Data Pipeline supporta i seguenti tipi di nodi di dati:

[DynamoDB DataNode](#)

Una tabella DynamoDB che contiene dati per [HiveActivity](#) o [EmrActivity](#) da utilizzare.

[SqlDataNode](#)

Una tabella SQL e una query di database che rappresentano i dati per un'attività di pipeline da utilizzare.

i Note

In precedenza, MySQLDataNode è stato utilizzato. Usare invece SqlDataNode.

[RedshiftDataNode](#)

Una tabella Amazon Redshift che contiene dati [RedshiftCopyActivity](#) da utilizzare.

[S3 DataNode](#)

Una posizione Amazon S3 che contiene uno o più file da utilizzare per un'attività di pipeline.

Database

AWS Data Pipeline supporta i seguenti tipi di database:

[JdbcDatabase](#)

Un database JDBC.

[RdsDatabase](#)

Un database Amazon RDS.

[RedshiftDatabase](#)

Un database Amazon Redshift.

Attività

In AWS Data Pipeline, un'attività è un componente della pipeline che definisce il lavoro da eseguire. AWS Data Pipeline fornisce diverse attività pre-configurate che supportano scenari comuni, ad esempio il trasferimento dei dati da una posizione a un'altra, l'esecuzione di query Hive e così via. Le attività sono estendibili, perciò è possibile eseguire script personalizzati per supportare infinite combinazioni.

AWS Data Pipeline supporta i seguenti tipi di attività:

[CopyActivity](#)

Copia i dati da una posizione a un'altra.

[EmrActivity](#)

Gestisce un cluster Amazon EMR.

[HiveActivity](#)

Esegue una query Hive su un cluster Amazon EMR.

[HiveCopyActivity](#)

Esegue una query Hive su un cluster Amazon EMR con supporto per il filtraggio avanzato dei dati e supporto per e. [S3 DataNode](#) [DynamoDB DataNode](#)

[PigActivity](#)

Esegue uno script Pig su un cluster Amazon EMR.

[RedshiftCopyActivity](#)

Copia i dati da e verso le tabelle Amazon Redshift.

[ShellCommandActivity](#)

Esegue un comando shell Linux/UNIX personalizzato come attività.

[SqlActivity](#)

Esegue una query SQL su un database.

Alcune attività hanno un supporto speciale per la gestione temporanea dei dati e delle tabelle di database. Per ulteriori informazioni, consulta [Dati e tabelle in gestione temporanea con attività della pipeline](#).

Precondizioni

In AWS Data Pipeline, una precondizione è un componente di una pipeline contenente istruzioni condizionali che devono essere true prima di eseguire un'attività. Ad esempio, una precondizione può verificare se i dati di origine sono presenti prima che un'attività della pipeline tenti di copiarli. AWS Data Pipeline fornisce diverse precondizioni predefinite che si adattano a scenari comuni, ad esempio l'esistenza di una tabella di database, la presenza di una chiave Amazon S3 e così via. Tuttavia, le precondizioni sono estendibili e consentono di eseguire script personalizzati a supporto di infinite combinazioni.

Sono disponibili due tipi di precondizioni: le precondizioni gestite dal sistema e le precondizioni gestite dall'utente. Le precondizioni gestite dal sistema vengono eseguite dal servizio Web AWS Data Pipeline a nome dell'utente e non richiedono una risorsa di calcolo. Le precondizioni gestite

dall'utente vengono eseguite solo sulla risorsa di calcolo che specifichi utilizzando i campi `runsOn` o `workerGroup`. La risorsa `workerGroup` deriva dall'attività che utilizza la preconditione.

Precondizioni gestite dal sistema

[DynamoDB DataExists](#)

Verifica se i dati esistono in una tabella DynamoDB specifica.

[DynamoDB TableExists](#)

Verifica se esiste una tabella DynamoDB.

[S3 KeyExists](#)

Verifica se esiste una chiave Amazon S3.

[S3 PrefixNotEmpty](#)

Verifica se un prefisso Amazon S3 è vuoto.

Precondizioni gestite dall'utente

[Exists](#)

Verifica se esiste un nodo di dati.

[ShellCommandPrecondition](#)

Esegue un comando shell Linux/Unix personalizzato come preconditione.

Risorse

In AWS Data Pipeline, una risorsa è la risorsa di elaborazione che esegue il lavoro che un'attività della pipeline specifica. AWS Data Pipeline supporta i seguenti tipi di risorse:

[Ec2Resource](#)

Un'istanza EC2 che esegue il lavoro definito da un'attività di pipeline.

[EmrCluster](#)

Un cluster Amazon EMR che esegue il lavoro definito da un'attività di pipeline, ad esempio.

[EmrActivity](#)

Le risorse possono essere eseguite nella stessa regione con i set di dati attivi e anche una regione diversa da AWS Data Pipeline. Per ulteriori informazioni, consulta [Utilizzo di una pipeline con risorse in più regioni](#).

Limiti delle risorse

AWS Data Pipeline ridimensiona per supportare un enorme numero di attività simultanee e può essere configurato per creare automaticamente le risorse necessarie per gestire carichi di lavoro di grandi dimensioni. Queste risorse create automaticamente sono sotto il controllo dell'utente e vengono conteggiate ai fini dei limiti delle risorse dell'account AWS. Ad esempio, se configuri la creazione automatica di un cluster Amazon EMR AWS Data Pipeline a 20 nodi per l'elaborazione dei dati e il tuo account AWS ha un limite di istanze EC2 impostato su 20, potresti inavvertitamente esaurire le risorse di backfill disponibili. Di conseguenza, è necessario considerare queste limitazioni in termini di risorse nel progetto oppure aumentare i limiti dell'account in base alle necessità. Per ulteriori informazioni sulle restrizioni dei servizi, consulta [Restrizioni dei servizi AWS](#) nella Guida di riferimento generale di AWS.

Note

Il limite è un'istanza per l'oggetto componente `Ec2Resource`.

Piattaforme supportate

Le pipeline possono avviare le tue risorse nelle seguenti piattaforme:

EC2-Classic

Le risorse vengono eseguite in una rete semplice, singola condivisa con altri clienti.

EC2-VPC

Le risorse vengono eseguite in un cloud privato virtuale (VPC, Virtual Private Cloud), logicamente limitato all'account AWS.

L'account AWS è in grado di avviare risorse in entrambe le piattaforme oppure solo in EC2-VPC, in base alle regioni. Per ulteriori informazioni, consulta la sezione [Piattaforme supportate](#) nella Guida per l'utente di Amazon EC2 per le istanze Linux.

Se l'account AWS supporta solo EC2-VPC, è necessario creare un VPC di default in ciascuna regione AWS. Per impostazione predefinita, è necessario avviare le proprie risorse in una sottorete di default del VPC predefinito. In alternativa, è possibile creare un VPC non predefinito e specificare una delle relative sottoreti quando si configurano le proprie risorse, quindi è necessario lanciare le proprie risorse nella sottorete specificata del VPC non predefinito.

Quando avvii un'istanza in un VPC, devi specificare un gruppo di sicurezza creato in modo specifico per quel VPC. Non è possibile specificare un gruppo di sicurezza creato per un EC2-Classical quando si avvia un'istanza in un VPC. Inoltre, è necessario utilizzare l'ID del gruppo di sicurezza e non il nome del gruppo di sicurezza per identificare un gruppo di sicurezza per un VPC.

Istanze Spot Amazon EC2 con cluster Amazon EMR e AWS Data Pipeline

Le pipeline possono utilizzare le istanze Spot di Amazon EC2 per i nodi di attività nelle risorse del cluster Amazon EMR. Per impostazione predefinita, le pipeline utilizzano le istanze on demand. Le istanze Spot consentono di usare le istanze EC2 inutilizzate ed eseguirle. Il modello di tariffazione delle istanze Spot è complementare a quello di istanze riservate e on demand, che possono potenzialmente offrire opzioni più convenienti per acquistare capacità di elaborazione, a seconda dell'applicazione per cui vengono impiegate. Per ulteriori informazioni, consulta la pagina di prodotto [Istanze Spot di Amazon EC2](#).

Quando utilizzi le istanze Spot, AWS Data Pipeline invia il prezzo massimo dell'istanza Spot ad Amazon EMR all'avvio del cluster. Applica automaticamente il lavoro del cluster al numero di nodi di task di istanze Spot definite utilizzando il campo `taskInstanceCount`. AWS Data Pipeline limita le istanze Spot per i nodi di task per assicurare che i nodi core on demand siano disponibili per eseguire la pipeline.

È possibile modificare un'istanza di risorse di pipeline non riuscita o completata per aggiungere le istanze Spot. Quando la pipeline lancia di nuovo il cluster, utilizza le istanze Spot per i nodi di task.

Considerazioni sulle istanze Spot

Quando si usano le istanze Spot con AWS Data Pipeline, si applicano le seguenti considerazioni:

- Le tue istanze Spot possono terminare quando il prezzo dell'istanza Spot supera il prezzo massimo dell'istanza o per motivi di capacità di Amazon EC2. Tuttavia, i dati non verranno persi in quanto AWS Data Pipeline impiega cluster con nodi core che sono sempre istanze on demand e non sono soggette a cessazione.

- Le istanze Spot possono richiedere più tempo per l'avvio in quanto soddisfano la capacità in modo asincrono. Pertanto, una pipeline di un'istanza Spot può essere eseguita più lentamente rispetto a una pipeline equivalente di un'istanza on demand.
- Il cluster potrebbe non essere eseguito se non si ricevono le istanze Spot, ad esempio nel caso in cui il prezzo massimo sia troppo basso.

Operazioni

Le azioni AWS Data Pipeline sono passaggi che il componente di una pipeline richiede quando si verificano determinati eventi, ad esempio attività con esito negativo e positivo o in ritardo. Il campo di eventi di un'attività si riferisce a un'operazione, ad esempio un riferimento a `snsAlarm` nel campo `onLateAction` di `EmrActivity`.

AWS Data Pipeline si basa sulle notifiche Amazon SNS come metodo principale per indicare lo stato delle pipeline e dei relativi componenti in modo automatico. Per ulteriori dettagli, consulta la pagina [Amazon SNS](#). Oltre alle notifiche SNS, è possibile utilizzare la console AWS Data Pipeline e l'interfaccia a riga di comando per ottenere informazioni sullo stato della pipeline.

AWS Data Pipeline supporta le seguenti operazioni:

[SnsAlarm](#)

Azione che invia una notifica SNS a un argomento basato sugli eventi `onSuccess`, `OnFail` e `onLateAction`.

[Interruzione](#)

Un'azione che attiva l'annullamento di un'attività in sospeso o non terminata, una risorsa o un nodo di dati. Non è possibile terminare azioni che includono `onSuccess`, `OnFail` o `onLateAction`.

Monitoraggio proattivo delle pipeline

Il modo migliore per rilevare problemi è monitorare le pipeline in modo proattivo sin dall'inizio. Puoi configurare i componenti della pipeline per informarti su determinate situazioni o eventi, ad esempio quando un componente della pipeline si guasta o non inizia entro l'ora di inizio pianificata. AWS Data Pipeline semplifica la configurazione delle notifiche fornendo campi evento sui componenti della pipeline che puoi associare alle notifiche di Amazon SNS, come `onSuccessOnFail`, e `onLateAction`.

Configurazione di AWS Data Pipeline

Prima di utilizzare AWS Data Pipeline per la prima volta, completa le seguenti attività.

Attività

- [Registrazione ad AWS](#)
- [Crea ruoli IAM per AWS Data Pipeline risorse e pipeline](#)
- [Consenti ai responsabili IAM \(utenti e gruppi\) di eseguire le azioni necessarie](#)
- [Concessione dell'accesso programmatico](#)

Dopo aver completato queste attività, è possibile iniziare a utilizzare AWS Data Pipeline. Per un tutorial di base, vedere [Nozioni di base su AWS Data Pipeline](#).

Registrazione ad AWS

Quando effettui la registrazione su Amazon Web Services (AWS), l'account AWS viene automaticamente registrato per tutti i servizi di AWS, compreso AWS Data Pipeline. Ti vengono addebitati solo i servizi che utilizzi. Per ulteriori informazioni sulle tariffe di utilizzo di AWS Data Pipeline, vedere [AWS Data Pipeline](#).

Registrarsi per creare un Account AWS

Se non disponi di un Account AWS, completa la procedura seguente per crearne uno.

Per registrarsi 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.

Durante la registrazione di un Account AWS, viene creato un Utente root dell'account AWS. L'utente root dispone dell'accesso a tutte le risorse e tutti i Servizi AWS nell'account. Come best practice di sicurezza, [assegna l'accesso amministrativo a un utente amministrativo](#) e utilizza solo l'utente root per eseguire [attività che richiedono l'accesso di un utente root](#).

Al termine del processo di registrazione, riceverai un'e-mail di conferma da AWS. 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.

Creazione di un utente amministratore

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.

Protezione dell'Utente root dell'account AWS

1. Accedi alla [AWS Management Console](#) come proprietario dell'account scegliendo Utente root e immettendo l'indirizzo email del Account AWS. Nella pagina successiva, inserisci la password.

Per informazioni sull'accesso utilizzando un utente root, consulta la pagina [Accesso come utente root](#) della Guida per l'utente di Accedi ad AWS.

2. Abilita l'autenticazione a più fattori (MFA) per l'utente root.

Per ricevere istruzioni, consulta [Abilitazione di un dispositivo MFA virtuale per l'utente root dell'Account AWS \(console\)](#) nella Guida per l'utente IAM.

Creazione di un utente amministratore

1. Abilita IAM Identity Center.

Per istruzioni, consulta [Enabling AWS IAM Identity Center](#) nella Guida AWS IAM Identity Center per l'utente.

2. In IAM Identity Center, concedi l'accesso amministrativo a un utente amministrativo.

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.

Accesso come utente amministratore

- 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 [Accedere al portale di accesso AWS](#) nella Guida per l'utente Accedi ad AWS.

Crea ruoli IAM per AWS Data Pipeline risorse e pipeline

AWS Data Pipeline richiede ruoli IAM che determinano le autorizzazioni per eseguire azioni e accedere alle AWS risorse. Il ruolo pipeline determina le autorizzazioni di cui AWS Data Pipeline dispone e un ruolo di risorsa determina le autorizzazioni di cui dispongono le applicazioni in esecuzione su risorse della pipeline, come le istanze EC2. Questi ruoli vengono specificati quando si crea una pipeline. Anche se non specificate un ruolo personalizzato e utilizzate i ruoli `DataPipelineDefaultRole` predefiniti `DataPipelineDefaultResourceRole`, dovete prima creare i ruoli e allegare le politiche di autorizzazione. Per ulteriori informazioni, consulta [Ruoli IAM per AWS Data Pipeline](#).

Consenti ai responsabili IAM (utenti e gruppi) di eseguire le azioni necessarie

Per lavorare con una pipeline, devi consentire a un responsabile IAM (un utente o un gruppo) del tuo account di eseguire [AWS Data Pipeline le azioni e le azioni](#) richieste per altri servizi, come definito dalla tua pipeline.

Per semplificare le autorizzazioni, è possibile allegare la policy `AWSDataPipeline_FullAccess` gestita ai principali IAM. Questa policy gestita consente al responsabile di eseguire tutte le azioni richieste dall'utente e l'`iam:PassRole` azione sui ruoli predefiniti utilizzati AWS Data Pipeline quando non è specificato un ruolo personalizzato.

Ti consigliamo vivamente di valutare attentamente questa politica gestita e di limitare le autorizzazioni solo a quelle richieste dagli utenti. Se necessario, utilizza questa policy come punto di partenza, quindi rimuovi le autorizzazioni per creare una policy di autorizzazioni in linea più restrittiva da collegare ai principali IAM. Per ulteriori informazioni ed esempi di politiche di autorizzazione, consulta [Esempi di policy per AWS Data Pipeline](#)

Una dichiarazione politica simile all'esempio seguente deve essere inclusa in una policy allegata a qualsiasi principale IAM che utilizza la pipeline. Questa dichiarazione consente al principale IAM di eseguire l'`iam:PassRole` azione sui ruoli utilizzati da una pipeline. Se non utilizzi ruoli predefiniti, sostituisci `MyPipelineRole` e `MyResourceRole` con i ruoli personalizzati che crei.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Action": "iam:PassRole",
      "Effect": "Allow",
      "Resource": [
        "arn:aws:iam::*:role/MyPipelineRole",
        "arn:aws:iam::*:role/MyResourceRole"
      ]
    }
  ]
}
```

La procedura seguente illustra come creare un gruppo IAM, allegare la policy `AWSDDataPipeline_FullAccess` gestita al gruppo e quindi aggiungere utenti al gruppo. È possibile utilizzare questa procedura per qualsiasi politica in linea

Per creare un gruppo di utenti **DataPipelineDevelopers** e allegare la politica `AWSDDataPipeline_FullAccess`

1. Aprire la console IAM all'indirizzo <https://console.aws.amazon.com/iam/>.
2. Nel riquadro di navigazione, seleziona Groups (Gruppi), Create New group (Crea nuovo gruppo).
3. Inserisci un nome di gruppo, ad esempio **DataPipelineDevelopers**, e quindi scegli Passaggio successivo.
4. Immettete **AWSDDataPipeline_FullAccess** Filtro, quindi selezionatelo dall'elenco.
5. Scegliere Next Step (Fase successiva), quindi scegliere Create Group (Crea gruppo).
6. Per aggiungere utenti al gruppo:
 - a. Seleziona il gruppo che hai creato dall'elenco dei gruppi.
 - b. Scegli Azioni di gruppo, Aggiungi utenti al gruppo.
 - c. Seleziona gli utenti che desideri aggiungere dall'elenco, quindi scegli Aggiungi utenti al gruppo.

Concessione dell'accesso programmatico

Gli utenti hanno bisogno di un accesso programmatico se desiderano interagire con AWS esternamente a AWS Management Console. La modalità con cui concedere l'accesso programmatico dipende dal tipo di utente che accede ad AWS.

Per fornire agli utenti l'accesso programmatico, scegli una delle seguenti opzioni.

Quale utente necessita dell'accesso programmatico?	Per	Come
Identità della forza lavoro (Utenti gestiti nel centro identità IAM)	Utilizza credenziali temporane e per firmare richieste programmatiche alla AWS CLI, agli SDK AWS o alle API AWS.	Segui le istruzioni per l'interfaccia che desideri utilizzare. <ul style="list-style-type: none"> • Per la AWS CLI, consulta la pagina Configurazione della AWS CLI per l'uso di AWS IAM Identity Center nella Guida per l'utente dell'AWS Command Line Interface. • Per gli SDK AWS, gli strumenti e le API AWS, consulta la pagina Autenticazione Centro identità IAM nella Guida di riferimento per SDK e strumenti AWS.
IAM	Utilizza credenziali temporane e per firmare richieste programmatiche alla AWS CLI, agli SDK AWS o alle API AWS.	Segui le istruzioni in Utilizzo di credenziali temporanee con le risorse AWS nella Guida per l'utente IAM.
IAM	(Non consigliato) Utilizza credenziali a lungo termine per firmare richieste programmatiche alla AWS	Segui le istruzioni per l'interfaccia che desideri utilizzare. <ul style="list-style-type: none"> • Per la AWS CLI, consulta la pagina Autenticazione

Quale utente necessita dell'accesso programmatico?	Per	Come
	CLI, agli SDK AWS o alle API AWS.	<p>tramite credenziali utente IAM nella Guida per l'utente dell'AWS Command Line Interface.</p> <ul style="list-style-type: none">• Per gli SDK e gli strumenti AWS, consulta la pagina Autenticazione con credenziali a lungo termine nella Guida di riferimento per SDK e strumenti AWS.• Per le API AWS, consulta la pagina Gestione delle chiavi di accesso per utenti IAM nella Guida per l'utente IAM.

Nozioni di base su AWS Data Pipeline

AWS Data Pipeline facilita la creazione di sequenze, la pianificazione, l'esecuzione e la gestione di carichi di lavoro per l'elaborazione di dati ricorrenti in modo affidabile e conveniente. Questo servizio semplifica la progettazione di attività extract-transform-load (ETL) utilizzando dati strutturati e non strutturati, sia in locale che nel cloud, in base alla logica aziendale.

Per utilizzare AWS Data Pipeline, è necessario creare una pipeline definition (definizione di pipeline) che specifica la logica di business per l'elaborazione dei dati. Una tipica definizione di pipeline è costituita da [attività](#) che definiscono il lavoro da eseguire e [nodi di dati](#) che definiscono la posizione e il tipo di dati di input e output.

In questo tutorial, si esegue lo script di un comando shell che conta il numero di richieste GET nei log del server Web Apache. Questa pipeline viene eseguita ogni 15 minuti per un'ora e scrive l'output su Amazon S3 a ogni iterazione.

Prerequisiti

Prima di iniziare, completa le attività in [Configurazione di AWS Data Pipeline](#).

Oggetti della pipeline

La pipeline utilizza i seguenti oggetti:

[ShellCommandActivity](#)

Legge i file di log di input e conta il numero di errori.

[S3 DataNode](#) (input)

Bucket S3 che contiene il file di log di input.

[S3 DataNode](#) (output)

Bucket S3 per l'output.

[Ec2Resource](#)

Le risorse di calcolo che AWS Data Pipeline utilizza per eseguire l'attività.

Si noti che se si dispone di una grande quantità di dati di file di log, è possibile configurare la pipeline per l'utilizzo di un cluster EMR e l'elaborazione dei file invece di un'istanza EC2.

Pianificazione

Stabilisce che l'attività venga eseguita ogni 15 minuti per un'ora.

Processi

- [Per creare la pipeline](#)
- [Monitorare la pipeline in esecuzione](#)
- [Visualizzazione dell'output](#)
- [Per eliminare la pipeline](#)

Per creare la pipeline

Il modo più rapido per iniziare con AWS Data Pipeline è utilizzare una definizione di pipeline denominata template (modello).

Per creare la pipeline

1. Apri la AWS Data Pipeline console all'[indirizzo https://console.aws.amazon.com/datapipeline/](https://console.aws.amazon.com/datapipeline/).
2. Nella barra di navigazione, selezionare una regione. È possibile selezionare qualsiasi regione disponibile, indipendentemente dalla posizione. Molte risorse AWS sono specifiche per una regione, ma AWS Data Pipeline consente di utilizzare le risorse in un'altra regione rispetto alla pipeline.
3. La prima schermata visualizzata dipende dal fatto che tu abbia creato una pipeline nella regione corrente.
 - a. Se non hai creato una pipeline in questa regione, la console visualizza una schermata introduttiva. Scegli Inizia subito.
 - b. Se hai già creato una pipeline in questa regione, la console visualizza una pagina che elenca le pipeline per la regione. Scegli Crea nuova pipeline.
4. In Nome, inserisci un nome per la pipeline.
5. (Facoltativo) In Descrizione, inserisci una descrizione per la pipeline.
6. Per Source, seleziona Crea utilizzando un modello, quindi seleziona il seguente modello: Guida introduttiva a utilizzare ShellCommandActivity.
7. Nella sezione Parameters (Parametri) che si è aperta quando è stato selezionato il modello, lasciare i valori predefiniti nella S3 input folder (cartella di input S3) e nel Shell command to

run (Comando Shell da eseguire). Fare clic sull'icona della cartella accanto a S3 output folder (Cartella di output S3), selezionare uno dei bucket o delle cartelle, quindi fare clic su Select (Seleziona).

8. In Schedule (Pianificazione), lasciare i valori predefiniti. Quando si attiva la pipeline, viene eseguito l'avvio della pipeline che si ripete ogni 15 minuti per un'ora.

Se si preferisce, è possibile selezionare Run once on pipeline activation (Esegui una volta all'attivazione della pipeline).

9. In Pipeline Configuration, lascia la registrazione abilitata. Scegli l'icona della cartella nella posizione S3 per i log, seleziona uno dei tuoi bucket o cartelle, quindi scegli Seleziona.

Se preferisci, puoi invece disattivare la registrazione.

10. In Security/Access, lascia i ruoli IAM impostati su Predefinito.

11. Fai clic su Activate (Attiva).

Se preferisci, puoi scegliere Modifica in Architect per modificare questa pipeline. Ad esempio, puoi aggiungere condizioni preliminari.

Monitorare la pipeline in esecuzione

Dopo aver attivato la pipeline, visualizzare la pagina Execution details (Dettagli esecuzione), dove è possibile monitorare l'avanzamento della pipeline.

Per monitorare l'avanzamento della pipeline

1. Fare clic su Update (Aggiorna) o premere F5 per aggiornare lo stato visualizzato.

Tip

Se non vi sono esecuzioni elencate, verificare che Start (in UTC) (Inizio (in UTC)) e End (in UTC) (Fine (in UTC)) coprano l'inizio e la fine pianificati della pipeline, quindi selezionare Update (Aggiorna).

2. Quando lo stato di ogni oggetto nella pipeline è FINISHED, significa che la tua pipeline ha completato correttamente le attività pianificate.

3. Se la pipeline non viene completata correttamente, verifica se vi sono problemi con le impostazioni della pipeline. Per ulteriori informazioni sulla risoluzione di problemi con istanze della pipeline non eseguite o non completate, consulta [Risoluzione dei problemi più comuni](#).

Visualizzazione dell'output

Apri la console Amazon S3 e accedi al tuo bucket. Se si esegue la pipeline ogni 15 minuti per un'ora, verranno visualizzate quattro sottocartelle con time-stamp. Ogni sottocartella contiene l'output in un file denominato `output.txt`. Poiché ogni volta lo script è stato eseguito sullo stesso file di input, i file di output sono identici.

Per eliminare la pipeline

Per non incorrere in addebiti, elimina la pipeline. L'eliminazione della pipeline comporta l'eliminazione della definizione della pipeline e di tutti gli oggetti associati.

Per eliminare la pipeline

1. Nella pagina Elenco tubazioni, seleziona la tua pipeline.
2. Fai clic su Azioni, quindi scegli Elimina.
3. Quando viene richiesta la conferma, seleziona Delete (Elimina).

Se hai finito con l'output di questo tutorial, elimina le cartelle di output dal tuo bucket Amazon S3.

Uso delle pipeline

È possibile amministrare, creare e modificare le pipeline utilizzando l'interfaccia a riga di comando (CLI) o l'SDK. AWS Le seguenti sezioni introducono concetti fondamentali AWS Data Pipeline e illustrano come lavorare con le pipeline.

Important

Prima di iniziare, consulta [Configurazione di AWS Data Pipeline](#).

Indice

- [Creare una pipeline](#)
- [Visualizzazione delle pipeline](#)
- [Modifica della pipeline](#)
- [Clonazione della pipeline](#)
- [Assegnazione di tag alla pipeline](#)
- [Disattivazione pipeline](#)
- [Eliminazione della pipeline](#)
- [Dati e tabelle in gestione temporanea con attività della pipeline](#)
- [Utilizzo di una pipeline con risorse in più regioni](#)
- [Guasti di una delle dipendenze e riesecuzioni](#)
- [Sintassi del file di definizione della pipeline](#)
- [Lavorare con l'API](#)

Creare una pipeline

AWS Data Pipeline offre vari modi per creare le pipeline:

- Usa AWS Command Line Interface (CLI) con un modello fornito per la tua comodità. Per ulteriori informazioni, consulta [Crea una pipeline dai modelli di Data Pipeline utilizzando la CLI](#).
- Utilizzare AWS Command Line Interface (CLI) con un file di definizione della pipeline in formato JSON.

- Utilizzare un SDK AWS con un'API specifica per il linguaggio. Per ulteriori informazioni, consulta [Lavorare con l'API](#).

Crea una pipeline dai modelli di Data Pipeline utilizzando la CLI

Data Pipeline fornisce diverse definizioni di pipeline preconfigurate, note come modelli. È possibile utilizzare i modelli per iniziare a utilizzare rapidamente AWS Data Pipeline. Questi modelli sono disponibili in un bucket pubblico presso la sede di Amazon S3: `s3://datapipeline-us-east-1/templates/` Questi modelli predefiniti sono creati per realizzare casi d'uso specifici e possono essere utilizzati per creare pipeline. Puoi usarlo `aws s3 ls --recursive "s3://datapipeline-us-east-1/templates/"` per elencare tutti i modelli disponibili.

Crea una pipeline da un modello utilizzando l'interfaccia a riga di comando

Supponiamo di voler creare una pipeline che esporti una tabella DynamoDB in Amazon S3. Il modello da utilizzare in questo caso è disponibile all'indirizzo: `s3://datapipeline-us-east-1/templates/DynamoDB Templates/Export DynamoDB table to S3.json`.

Per scaricare il modello JSON e creare una pipeline utilizzando la CLI

1. Scarica il modello utilizzando la `aws s3 cp` CLI o il `curl`. Ad esempio:

```
aws s3 cp "s3://datapipeline-us-east-1/templates/DynamoDB Templates/Export DynamoDB table to S3.json" <destination directory>
```

2. Apporta le modifiche necessarie al modello scaricato. Ad esempio, per utilizzare la versione più recente di EMR, modifica il `releaseLabel` campo nell'`EmrClusterForBackup` oggetto, modifica i tipi di istanza principale e principale e modifica i valori predefiniti dei parametri nel modello.
3. Crea una pipeline utilizzando l'interfaccia a `create-pipeline` riga di comando. Ad esempio:

```
aws datapipeline create-pipeline --name my-ddb-backup-pipeline --unique-id my-ddb-backup-pipeline --region ap-northeast-1
```

4. Annotate l'ID della pipeline creata.
5. Da `put-pipeline-definition` utilizzare per caricare la definizione. Fornisci i valori dei parametri di cui desideri sovrascrivere i valori predefiniti utilizzando l'`--parameter-values` opzione.

Per ulteriori informazioni sui modelli, consulta [Choose a template \(Scegli un modello\)](#).

Choose a template (Scegli un modello)

I seguenti modelli sono disponibili per il download dal bucket Amazon S3: `s3://datapipeline-us-east-1/templates/`

Modelli di

- [Nozioni di base sull'uso di ShellCommandActivity](#)
- [Esegui il AWS comando CLI](#)
- [Esporta la tabella DynamoDB in S3](#)
- [Importa i dati di backup di DynamoDB da S3](#)
- [Esegui un processo su un cluster Amazon EMR](#)
- [Copia completa di Amazon RDS MySQL Table su Amazon S3](#)
- [Copia incrementale della tabella MySQL di Amazon RDS su Amazon S3](#)
- [Carica i dati S3 nella tabella MySQL di Amazon RDS](#)
- [Copia completa della tabella MySQL di Amazon RDS su Amazon Redshift](#)
- [Copia incrementale di una tabella MySQL di Amazon RDS su Amazon Redshift](#)
- [Carica dati da Amazon S3 in Amazon Redshift](#)

Nozioni di base sull'uso di ShellCommandActivity

Il ShellCommandActivity modello Getting Started using esegue uno script di comandi shell per contare il numero di richieste GET in un file di registro. L'output viene scritto in una posizione Amazon S3 con data e ora su ogni esecuzione pianificata della pipeline.

Il modello utilizza i seguenti oggetti della pipeline:

- ShellCommandActivity
- S3 InputNode
- S3 OutputNode
- Ec2Resource

Esegui il AWS comando CLI

Questo modello esegue un comando AWS CLI definito dall'utente a intervalli pianificati.

Esporta la tabella DynamoDB in S3

Il modello Esporta tabella DynamoDB in S3 pianifica un cluster Amazon EMR per esportare dati da una tabella DynamoDB a un bucket Amazon S3. Questo modello utilizza un cluster Amazon EMR, che è dimensionato proporzionalmente al valore del throughput disponibile nella tabella DynamoDB. Sebbene sia possibile aumentare IOPS su una tabella, questo potrebbe comportare costi supplementari durante l'importazione e l'esportazione. In precedenza, l'esportazione utilizzava un file nativoHiveActivity, ma ora utilizza nativoMapReduce.

Il modello utilizza i seguenti oggetti della pipeline:

- [EmrActivity](#)
- [EmrCluster](#)
- [DynamoDB DataNode](#)
- [S3 DataNode](#)

Importa i dati di backup di DynamoDB da S3

Il modello Importa dati di backup DynamoDB da S3 pianifica un cluster Amazon EMR per caricare un backup DynamoDB creato in precedenza in Amazon S3 su una tabella DynamoDB. Gli elementi esistenti nella tabella DynamoDB vengono aggiornati con quelli dei dati di backup e nuovi elementi vengono aggiunti alla tabella. Questo modello utilizza un cluster Amazon EMR, che è dimensionato proporzionalmente al valore del throughput disponibile nella tabella DynamoDB. Sebbene sia possibile aumentare IOPS su una tabella, questo potrebbe comportare costi supplementari durante l'importazione e l'esportazione. In precedenza, l'importazione utilizzava un file nativoHiveActivity, ma ora utilizza nativoMapReduce.

Il modello utilizza i seguenti oggetti della pipeline:

- [EmrActivity](#)
- [EmrCluster](#)
- [DynamoDB DataNode](#)
- [S3 DataNode](#)

- [S3 PrefixNotEmpty](#)

Esegui un processo su un cluster Amazon EMR

Il modello Run Job on an Elastic MapReduce Cluster avvia un cluster Amazon EMR in base ai parametri forniti e avvia i passaggi in base alla pianificazione specificata. Una volta completato il processo, il cluster EMR viene terminato. Le operazioni di bootstrap opzionali possono essere specificate per installare software aggiuntivo o modificare la configurazione di applicazioni nel cluster.

Il modello utilizza i seguenti oggetti della pipeline:

- [EmrActivity](#)
- [EmrCluster](#)

Copia completa di Amazon RDS MySQL Table su Amazon S3

Il modello Copia completa di RDS MySQL Table to S3 copia un'intera tabella Amazon RDS MySQL e archivia l'output in una posizione Amazon S3. L'output viene archiviato come file CSV in una sottocartella con data e ora nella posizione Amazon S3 specificata.

Il modello utilizza i seguenti oggetti della pipeline:

- [CopyActivity](#)
- [Ec2Resource](#)
- [SqlDataNode](#)
- [S3 DataNode](#)

Copia incrementale della tabella MySQL di Amazon RDS su Amazon S3

Il modello Incremental Copy of RDS MySQL Table to S3 esegue una copia incrementale dei dati da una tabella MySQL Amazon RDS e archivia l'output in una posizione Amazon S3. La tabella MySQL di Amazon RDS deve avere una colonna Ultima modifica.

Questo modello copia le modifiche apportate alla tabella tra intervalli pianificati a partire dalla data di inizio pianificata. Il tipo di pianificazione è una serie temporale, quindi se una copia è stata programmata per una determinata ora, AWS Data Pipeline copia le righe della tabella con un timestamp di Ultima modifica compreso nell'ora. Le eliminazioni fisiche fatte alla tabella non vengono

copiate. L'output viene scritto in una sottocartella con data e ora nella posizione Amazon S3 a ogni esecuzione pianificata.

Il modello utilizza i seguenti oggetti della pipeline:

- [CopyActivity](#)
- [Ec2Resource](#)
- [SqlDataNode](#)
- [S3 DataNode](#)

Carica i dati S3 nella tabella MySQL di Amazon RDS

Il modello Load S3 Data into RDS MySQL Table pianifica un'istanza Amazon EC2 per copiare il file CSV dal percorso del file Amazon S3 specificato di seguito in una tabella Amazon RDS MySQL. Il file CSV non deve avere una riga di intestazione. Il modello aggiorna le voci esistenti nella tabella MySQL di Amazon RDS con quelle nei dati di Amazon S3 e aggiunge nuove voci dai dati di Amazon S3 alla tabella MySQL di Amazon RDS. È possibile caricare i dati in una tabella esistente o fornire una query SQL per creare una nuova tabella.

Il modello utilizza i seguenti oggetti della pipeline:

- [CopyActivity](#)
- [Ec2Resource](#)
- [SqlDataNode](#)
- [S3 DataNode](#)

Modelli da Amazon RDS ad Amazon Redshift

I due modelli seguenti copiano le tabelle da Amazon RDS MySQL ad Amazon Redshift utilizzando uno script di traduzione, che crea una tabella Amazon Redshift utilizzando lo schema della tabella di origine con le seguenti avvertenze:

- Se non viene specificata una chiave di distribuzione, la prima chiave primaria della tabella Amazon RDS viene impostata come chiave di distribuzione.
- Non è possibile saltare una colonna presente in una tabella MySQL di Amazon RDS quando si esegue una copia su Amazon Redshift.

- (Facoltativo) Puoi fornire una mappatura dei tipi di dati delle colonne da Amazon RDS MySQL ad Amazon Redshift come uno dei parametri del modello. Se specificato, lo script lo utilizza per creare la tabella Amazon Redshift.

Se viene utilizzata la modalità di inserimento `Overwrite_Existing` Amazon Redshift:

- Se non viene fornita una chiave di distribuzione, viene utilizzata una chiave primaria nella tabella MySQL di Amazon RDS.
- Se ci sono chiavi primarie composite nella tabella, la prima viene usata come chiave di distribuzione se la chiave di distribuzione non viene fornita. Solo la prima chiave composita viene impostata come chiave primaria nella tabella Amazon Redshift.
- Se non viene fornita una chiave di distribuzione e non è presente alcuna chiave primaria nella tabella MySQL di Amazon RDS, l'operazione di copia ha esito negativo.

Per ulteriori informazioni su Amazon Redshift, consulta i seguenti argomenti:

- [Cluster Amazon Redshift](#)
- [Copia Amazon Redshift](#)
- [Stili di distribuzione](#) ed [esempi](#) DISTKEY
- [Chiavi ordinamento](#)

La tabella seguente spiega come lo script traduce i tipi di dati:

Traduzioni dei tipi di dati tra MySQL e Amazon Redshift

Tipi di dati MySQL	Tipo di dati Amazon Redshift	Note
TINYINT, TINYINT (dimensioni)	SMALLINT	MySQL: -da 128 a 127. Il numero massimo di cifre può essere specificato tra parentesi. Amazon Redshift: INT2. Intero a due byte firmato
TINYINT UNSIGNED,	SMALLINT	MySQL: da 0 a 255 UNSIGNED. Il numero

Tipi di dati MySQL	Tipo di dati Amazon Redshift	Note
TINYINT (dimensione) UNSIGNED		<p>massimo di cifre può essere specificato tra parentesi.</p> <p>Amazon Redshift: INT2. Intero a due byte firmato</p>
SMALLINT, SMALLINT (dimensioni)	SMALLINT	<p>MySQL: - da 32768 a 32767 normale. Il numero massimo di cifre può essere specificato tra parentesi.</p> <p>Amazon Redshift: INT2. Intero a due byte firmato</p>
SMALLINT UNSIGNED, SMALLINT(dimensione) UNSIGNED,	INTEGER	<p>MySQL: da 0 a 65535 UNSIGNED*. Il numero massimo di cifre può essere specificato tra parentesi</p> <p>Amazon Redshift: INT4. Intero a quattro byte firmato</p>
MEDIUMINT, MEDIUMINT(dimensione)	INTEGER	<p>MySQL: da 388608 a 8388607. Il numero massimo di cifre può essere specificato tra parentesi</p> <p>Amazon Redshift: INT4. Intero a quattro byte firmato</p>
MEDIUMINT UNSIGNED, MEDIUMINT(dimensione) UNSIGNED	INTEGER	<p>MySQL: da 0 a 16777215. Il numero massimo di cifre può essere specificato tra parentesi</p> <p>Amazon Redshift: INT4. Intero a quattro byte firmato</p>

Tipi di dati MySQL	Tipo di dati Amazon Redshift	Note
INT, INT(dimensione)	INTEGER	MySQL: da 147483648 a 2147483647 Amazon Redshift: INT4. Intero a quattro byte firmato
INT UNSIGNED, INT(dimensione) UNSIGNED	BIGINT	MySQL: da 0 a 4294967295 Amazon Redshift: INT8. Intero a otto byte firmato
BIGINT BIGINT (dimensione)	BIGINT	Amazon Redshift: INT8. Intero a otto byte firmato
BIGINT UNSIGNED BIGINT(dimensione) UNSIGNED	VARCHAR(20*4)	MySQL: da 0 a 184467440 73709551615 Amazon Redshift: nessun equivalente nativo, quindi utilizzo di un array di caratteri.
FLOAT FLOAT (dimensioni, d) FLOAT (dimensioni, d) UNSIGNED	REAL	Il numero massimo di cifre può essere specificato nel parametro della dimensione. Il numero di cifre alla destra del punto decimale viene specificato nel parametro d. Amazon Redshift: FLOAT4

Tipi di dati MySQL	Tipo di dati Amazon Redshift	Note
DOUBLE (dimensioni, d)	DOUBLE PRECISION	<p>Il numero massimo di cifre può essere specificato nel parametro della dimensione. Il numero di cifre alla destra del punto decimale viene specificato nel parametro d.</p> <p>Amazon Redshift: FLOAT8</p>
DECIMAL (dimensioni, d)	DECIMAL (dimensioni, d)	<p>Un DOUBLE memorizzato come stringa, consentendo una virgola decimale fissa. Il numero massimo di cifre può essere specificato nel parametro della dimensione. Il numero di cifre alla destra del punto decimale viene specificato nel parametro d.</p> <p>Amazon Redshift: nessun equivalente nativo.</p>

Tipi di dati MySQL	Tipo di dati Amazon Redshift	Note
CHAR(dimensione)	VARCHAR (dimensione* 4)	<p>Contiene una stringa di lunghezza fissa, che può contenere lettere, numeri e caratteri speciali. La dimensione fissa viene specificata come parametro tra parentesi. Può contenere fino a 255 caratteri.</p> <p>Riempito a destra con spazi.</p> <p>Amazon Redshift: il tipo di dati CHAR non supporta caratteri multibyte, quindi viene utilizzato o VARCHAR.</p> <p>Il numero massimo di byte per carattere è 4 secondo RFC3629, che limita la tabella dei caratteri a U+10FFFF.</p>
VARCHAR(dimensione)	VARCHAR (dimensione* 4)	<p>Può contenere fino a 255 caratteri.</p> <p>VARCHAR non supporta i seguenti punti di codice UTF-8 non validi: 0xD800- 0xDFFF, (sequenze in byte: ED A0 80- ED BF BF), 0xFDD0- 0xFDEF, 0xFFFE e 0xFFFF (sequenze in byte: EF B7 90- EF B7 AF, EF BF BE e EF BF BF)</p>
TINYTEXT	VARCHAR(255*4)	<p>Contiene una stringa con un massimo di 255 caratteri</p>

Tipi di dati MySQL	Tipo di dati Amazon Redshift	Note
TEXT	VARCHAR(max)	Contiene una stringa con un massimo di 65.535 caratteri.
MEDIUMTEXT	VARCHAR(max)	Da 0 a 16.777.215 char
LONGTEXT	VARCHAR(max)	Da 0 a 4.294.967.295 char
BOOLEAN BOOL TINYINT(1)	BOOLEAN	MySQL: questi tipi sono sinonimi di TINYINT (1) . Il valore zero è considerato falso. I valori diversi da zero sono considerati veri.
BINARY[(M)]	varchar(255)	M è da 0 a 255 byte, FIXED
VARBINARY(M)	VARCHAR(max)	Da 0 a 65.535 byte
TINYBLOB	VARCHAR(255)	Da 0 a 255 byte
BLOB	VARCHAR(max)	Da 0 a 65.535 byte
MEDIUMBLOB	VARCHAR(max)	Da 0 a 16.777.215 byte
LOB	VARCHAR(max)	Da 0 a 4.294.967.295 byte
ENUM	VARCHAR(255*2)	Il limite non è alla lunghezza della stringa di enumerazione letterale, bensì sulla definizione della tabella per il numero di valori di enumerazione.
SET	VARCHAR(255*2)	Come enum.
DATE	DATE	(YYYY-MM-DD) da "1000-01-01" a "9999-12-31"

Tipi di dati MySQL	Tipo di dati Amazon Redshift	Note
TIME	VARCHAR(10*4)	(hh:mm:ss) da "-838:59:59" a "838:59:59"
DATETIME	TIMESTAMP	(YYYY-MM-DD hh:mm:ss) da "1000-01-01 00:00:00" a "9999-12-31 23:59:59"
TIMESTAMP	TIMESTAMP	(YYYYMMDDhhmmss) da 19700101000000 a 2037+
ANNO	VARCHAR(4*4)	(YYYY) Da 1900 a 2155
colonna SERIAL	<p>generazione di ID/Questo attributo non è necessario per un data warehouse OLAP poiché questa colonna è copiata.</p> <p>La parola chiave SERIAL non viene aggiunta durante la traduzione.</p>	<p>SERIAL è un'entità denominata SEQUENCE. Esiste indipendentemente dal resto della tabella.</p> <p>colonna GENERATED BY DEFAULT</p> <p>equivalente a:</p> <p>Nome CREATE SEQUENCE; tabella CREATE TABLE (colonna INTEGER NOT NULL DEFAULT nextval (name));</p>

Tipi di dati MySQL	Tipo di dati Amazon Redshift	Note
colonna BIGINT UNSIGNED NOT NULL AUTO_INCREMENT UNIQUE	<p>generazione di ID/Questo attributo non è necessario per un data warehouse OLAP poiché questa colonna è copiata.</p> <p>Quindi, la parola chiave SERIAL non viene aggiunta durante la traduzione.</p>	<p>SERIAL è un'entità denominata SEQUENCE. Esiste indipendentemente dal resto della tabella.</p> <p>colonna GENERATED BY DEFAULT equivalente a:</p> <p>Nome CREATE SEQUENCE; tabella CREATE TABLE (colonna INTEGER NOT NULL DEFAULT nextval (name));</p>
ZEROFILL	La parola chiave ZEROFILL non viene aggiunta durante la traduzione.	<p>INT UNSIGNED ZEROFILL NOT NULL</p> <p>ZEROFILL riempie il valore visualizzato del campo con zeri fino alla larghezza di visualizzazione specificata nella definizione della colonna. I valori superiori alla larghezza di visualizzazione non sono troncati. Si noti che l'utilizzo di ZEROFILL implica anche UNSIGNED.</p>

Copia completa della tabella MySQL di Amazon RDS su Amazon Redshift

Il modello Copia completa della tabella Amazon RDS MySQL su Amazon Redshift copia l'intera tabella Amazon RDS MySQL in una tabella Amazon Redshift archiviando i dati in una cartella Amazon S3. La cartella di staging Amazon S3 deve trovarsi nella stessa regione del cluster Amazon Redshift. Una tabella Amazon Redshift viene creata con lo stesso schema della tabella Amazon RDS

MySQL di origine, se non esiste già. Fornisci qualsiasi tipo di dati di colonna da Amazon RDS MySQL ad Amazon Redshift che desideri applicare durante la creazione di una tabella Amazon Redshift.

Il modello utilizza i seguenti oggetti della pipeline:

- [CopyActivity](#)
- [RedshiftCopyActivity](#)
- [S3 DataNode](#)
- [SqlDataNode](#)
- [RedshiftDataNode](#)
- [RedshiftDatabase](#)

Copia incrementale di una tabella MySQL di Amazon RDS su Amazon Redshift

La copia incrementale della tabella Amazon RDS MySQL nel modello Amazon Redshift copia i dati da una tabella Amazon RDS MySQL a una tabella Amazon Redshift archiviando i dati in una cartella Amazon S3.

La cartella di staging Amazon S3 deve trovarsi nella stessa regione del cluster Amazon Redshift.

AWS Data Pipeline utilizza uno script di traduzione per creare una tabella Amazon Redshift con lo stesso schema della tabella MySQL Amazon RDS di origine, se non esiste già. Devi fornire qualsiasi tipo di dati di colonna da Amazon RDS MySQL ad Amazon Redshift che desideri applicare durante la creazione di una tabella Amazon Redshift.

Questo modello copia le modifiche apportate alla tabella MySQL di Amazon RDS tra intervalli pianificati, a partire dall'ora di inizio pianificata. Le eliminazioni fisiche nella tabella MySQL di Amazon RDS non vengono copiate. È necessario specificare il nome della colonna che memorizza l'ultimo valore temporale modificato.

Quando utilizzi il modello predefinito per creare pipeline per la copia incrementale di Amazon RDS, viene creata un'attività con il nome `RDSToS3CopyActivity` predefinito. È possibile assegnarle un nome diverso.

Il modello utilizza i seguenti oggetti della pipeline:

- [CopyActivity](#)

- [RedshiftCopyActivity](#)
- [S3 DataNode](#)
- [SqlDataNode](#)
- [RedshiftDataNode](#)
- [RedshiftDatabase](#)

Carica dati da Amazon S3 in Amazon Redshift

Il modello Load data from S3 into Redshift copia i dati da una cartella Amazon S3 in una tabella Amazon Redshift. È possibile caricare i dati in una tabella esistente o fornire una query SQL per creare una tabella.

I dati vengono copiati in base alle opzioni di Amazon RedshiftCOPY. La tabella Amazon Redshift deve avere lo stesso schema dei dati in Amazon S3. Per COPY le opzioni, consulta [COPY](#) nella Amazon Redshift Database Developer Guide.

Il modello utilizza i seguenti oggetti della pipeline:

- [CopyActivity](#)
- [RedshiftCopyActivity](#)
- [S3 DataNode](#)
- [RedshiftDataNode](#)
- [RedshiftDatabase](#)
- [Ec2Resource](#)

Creazione di una pipeline utilizzando modelli parametrizzati

È possibile utilizzare un modello parametrizzato per personalizzare una definizione di pipeline. In questo modo è possibile creare una definizione di pipeline comune offrendo diversi parametri quando si aggiunge la definizione di pipeline a una nuova pipeline.

Indice

- [Aggiungi myVariables alla definizione della pipeline](#)
- [Definire oggetti parametrici](#)

- [Definire i valori di parametro](#)
- [Invio della definizione della pipeline](#)

Aggiungi myVariables alla definizione della pipeline

Quando si crea il file di definizione della pipeline, specificare le variabili utilizzando la sintassi seguente: `#{myVariable}`. È necessario che la variabile abbia il prefisso my. *Ad esempio, il seguente file di definizione della pipeline include le seguenti variabili: myShellCmdMyS3 e MyS3 InputLoc. pipeline-definition.json OutputLoc*

Note

Una definizione di pipeline dispone di un limite massimo di 50 parametri.

```
{
  "objects": [
    {
      "id": "ShellCommandActivityObj",
      "input": {
        "ref": "S3InputLocation"
      },
      "name": "ShellCommandActivityObj",
      "runsOn": {
        "ref": "EC2ResourceObj"
      },
      "command": " #{myShellCmd}",
      "output": {
        "ref": "S3OutputLocation"
      },
      "type": "ShellCommandActivity",
      "stage": "true"
    },
    {
      "id": "Default",
      "scheduleType": "CRON",
      "failureAndRerunMode": "CASCADE",
      "schedule": {
        "ref": "Schedule_15mins"
      },
      "name": "Default",
```

```

    "role": "DataPipelineDefaultRole",
    "resourceRole": "DataPipelineDefaultResourceRole"
  },
  {
    "id": "S3InputLocation",
    "name": "S3InputLocation",
    "directoryPath": "#{myS3InputLoc}",
    "type": "S3DataNode"
  },
  {
    "id": "S3OutputLocation",
    "name": "S3OutputLocation",
    "directoryPath": "#{myS3OutputLoc}/#{format(@scheduledStartTime, 'YYYY-MM-dd-HH-mm-ss')}",
    "type": "S3DataNode"
  },
  {
    "id": "Schedule_15mins",
    "occurrences": "4",
    "name": "Every 15 minutes",
    "startAt": "FIRST_ACTIVATION_DATE_TIME",
    "type": "Schedule",
    "period": "15 Minutes"
  },
  {
    "terminateAfter": "20 Minutes",
    "id": "EC2ResourceObj",
    "name": "EC2ResourceObj",
    "instanceType": "t1.micro",
    "type": "Ec2Resource"
  }
]
}

```

Definire oggetti parametrici

È possibile creare un file separato con gli oggetti dei parametri che definisca le variabili della definizione della pipeline. Ad esempio, il seguente file JSON contiene oggetti `parameters.json` parametrici per le `OutputLoc` variabili `myShellCmd`, `MyS3` e `MyS3` della `InputLoc` definizione della pipeline di esempio sopra riportata.

```

{
  "parameters": [

```

```

{
  "id": "myShellCmd",
  "description": "Shell command to run",
  "type": "String",
  "default": "grep -rc \"GET\" ${INPUT1_STAGING_DIR}/* > ${OUTPUT1_STAGING_DIR}/
output.txt"
},
{
  "id": "myS3InputLoc",
  "description": "S3 input location",
  "type": "AWS::S3::ObjectKey",
  "default": "s3://us-east-1.elasticmapreduce.samples/pig-apache-logs/data"
},
{
  "id": "myS3OutputLoc",
  "description": "S3 output location",
  "type": "AWS::S3::ObjectKey"
}
]
}

```

Note

È possibile aggiungere questi oggetti direttamente al file di definizione della pipeline invece di utilizzare un file separato.

La tabella seguente descrive gli attributi per gli oggetti dei parametri.

Attributi dei parametri

Attributo	Type (Tipo)	Descrizione
id	Stringa	Identificatore univoco del parametro. Per mascherare il valore mentre è digitato o visualizzato, aggiungere un asterisco (*) come prefisso. Ad esempio, *myVariable —. Da notare che questo, inoltre, crittografa il valore prima di

Attributo	Type (Tipo)	Descrizione
		essere memorizzato da AWS Data Pipeline.
description	Stringa	Descrizione del parametro.
type	Stringa, numero intero, doppio o AWS::S3::ObjectKey	Il tipo di parametro che definisce l'intervallo consentito di valori di input e regole di convalida. L'impostazione predefinita è Stringa.
facoltativo	Boolean	Indica se il parametro è obbligatorio o facoltativo. Il valore predefinito è false.
allowedValues	Elenco di stringhe	Enumera tutti i valori consentiti per il parametro.
default	Stringa	Il valore predefinito per il parametro. Se si specifica un valore per questo parametro utilizzando i valori dei parametri, sostituisce il valore di default.
isArray	Boolean	Indica se il parametro è un array.

Definire i valori di parametro

È possibile creare un file separato per definire le variabili utilizzando i valori dei parametri. Ad esempio, il seguente file JSON contiene il valore per la OutputLoc variabile *MyS3* dalla definizione di pipeline di esempio sopra riportata. file://values.json

```
{
  "values":
  {
```

```
    "myS3OutputLoc": "myOutputLocation"  
  }  
}
```

Invio della definizione della pipeline

Quando si invia la definizione di pipeline, è possibile specificare i parametri, gli oggetti dei parametri e i valori dei parametri. Ad esempio, puoi usare il [put-pipeline-definition](#) AWS CLI comando come segue:

```
$ aws datapipeline put-pipeline-definition --pipeline-id id --pipeline-definition  
file://pipeline-definition.json \  
--parameter-objects file://parameters.json --parameter-values-uri file://values.json
```

Note

Una definizione di pipeline dispone di un limite massimo di 50 parametri. La dimensione del file per `parameter-values-uri` dispone di un limite massimo di 15 KB.

Visualizzazione delle pipeline

Puoi visualizzare le tue pipeline utilizzando l'interfaccia a riga di comando (CLI).

Per visualizzare le pipeline utilizzando la AWS CLI

- Utilizzare il comando [list-pipelines](#) per elencare le pipeline:

```
aws datapipeline list-pipelines
```

Interpretazione dei codici sullo stato della pipeline

I livelli di stato visualizzati nella console AWS Data Pipeline e nell'interfaccia a riga di comando indicano la condizione di una pipeline e dei relativi componenti. Lo stato della pipeline è semplicemente una panoramica di una pipeline; per visualizzare ulteriori informazioni, visualizzare lo stato dei singoli componenti della pipeline.

Una pipeline ha uno stato SCHEDULED se è pronto (la definizione di pipeline ha passato la convalida), al momento è in esecuzione il lavoro, oppure l'esecuzione del lavoro è terminata. Una pipeline con lo

stato PENDING se non è attivato o non è in grado di eseguire il lavoro (ad esempio, la convalida della definizione della pipeline non è riuscita).

Una pipeline è considerata inattiva se il suo stato è PENDING, INACTIVE o FINISHED. Alle pipeline inattive verrà applicato un costo (per ulteriori informazioni, consulta la pagina [Prezzi](#)).

Codici di stato

ACTIVATING

Il componente o la risorsa è in fase di avvio, ad esempio un'istanza EC2.

CANCELED

Il componente è stato annullato da un utente o AWS Data Pipeline prima che potesse essere eseguito. Ciò può accadere automaticamente quando si verifica un errore in un componente o in una risorsa diversa da cui dipende questo componente.

CASCADE_FAILED

Il componente o la risorsa è stato annullato a causa di un errore a cascata proveniente da una delle sue dipendenze, ma probabilmente il componente non era l'origine originale dell'errore.

DEACTIVATING

La pipeline è in fase di disattivazione.

FAILED

Il componente o la risorsa ha riscontrato un errore e ha smesso di funzionare. Quando un componente o una risorsa si guasta, è possibile che le cancellazioni e gli errori si riversino a cascata su altri componenti che ne dipendono.

FINISHED

Il componente ha completato il lavoro assegnato.

INACTIVE

La pipeline è stata disattivata.

PAUSED

Il componente è stato messo in pausa e non sta attualmente eseguendo il suo lavoro.

PENDING

La pipeline è pronta per essere attivata per la prima volta.

RUNNING

La risorsa è in esecuzione e pronta per ricevere lavoro.

SCHEDULED

L'esecuzione della risorsa è pianificata.

SHUTTING_DOWN

La risorsa si sta spegnendo dopo aver completato con successo il suo lavoro.

SKIPPED

Il componente ha saltato gli intervalli di esecuzione dopo l'attivazione della pipeline utilizzando un timestamp successivo alla pianificazione corrente.

TIMEDOUT

La risorsa ha superato la `terminateAfter` soglia ed è stata interrotta da AWS Data Pipeline. Dopo che la risorsa ha raggiunto questo stato, AWS Data Pipeline ignora i `actionOnResourceFailure` `retryTimeout` valori e e di quella risorsa. `retryDelay` Questo stato si applica solo alle risorse.

VALIDATING

La definizione della pipeline viene convalidata da AWS Data Pipeline.

WAITING_FOR_RUNNER

Il componente è in attesa che il suo cliente lavoratore recuperi un elemento di lavoro. La relazione tra componente e lavoratore cliente è controllata dai `workerGroup` campi `runsOn` or definiti da quel componente.

WAITING_ON_DEPENDENCIES

Il componente sta verificando che i prerequisiti predefiniti e configurati dall'utente siano soddisfatti prima di eseguire il suo lavoro.

Interpretazione dello stato di pipeline e componenti

Ogni pipeline e componente all'interno di quella pipeline restituisce uno stato di integrità di HEALTHY, ERROR, "-", No Completed Executions o No Health Information Available. Una pipeline dispone di un solo stato di integrità dopo che un componente della pipeline ha completato la prima esecuzione o se le precondizioni del componente hanno dato esito negativo. Lo stato di

integrità per i componenti aggregati in uno stato di integrità della pipeline negli stati di errore è visibile prima, al momento della visualizzazione dei dettagli di esecuzione.

Stati di integrità delle pipeline

HEALTHY

Lo stato di integrità aggregato di tutti i componenti è HEALTHY. Questo significa che almeno un componente deve essere completato con successo. È possibile fare clic sullo stato HEALTHY per vedere l'istanza dei componenti della pipeline più recente completata con successo nella pagina di Execution Details (Dettagli di esecuzione).

ERROR

Almeno un componente della pipeline dispone di un stato di integrità di ERROR. È possibile fare clic sullo stato ERROR per vedere l'istanza dei componenti della pipeline fallita più di recente nella pagina di Execution Details (Dettagli di esecuzione).

No Completed Executions o No Health Information Available.

Nessun stato di integrità è stato segnalato per questa pipeline.

Note

Mentre i componenti aggiornano lo stato di integrità quasi immediatamente, possono essere necessari fino a cinque minuti per aggiornare lo stato di integrità di una pipeline.

Stati di integrità dei componenti

HEALTHY

Un componente (Activity o DataNode) ha uno stato di integrità di HEALTHY nel caso in cui una esecuzione viene completata con successo, contrassegnata con stato FINISHED o MARK_FINISHED. È possibile fare clic sul nome del componente o sullo stato HEALTHY per vedere le istanze dei componenti della pipeline più recenti completati con successo nella pagina di Execution Details (Dettagli di esecuzione).

ERROR

Si è verificato un errore a livello di componente o una delle precondizioni non è riuscita. Gli stati FAILED, TIMEOUT oppure CANCELED attivano questo errore. È possibile fare clic sul nome del

componente o sullo stato ERROR per vedere le istanze dei componenti della pipeline più recenti non riusciti nella pagina di Execution Details (Dettagli di esecuzione).

No Completed Executions o No Health Information Available

Nessun stato di integrità è stato segnalato per questo componente.

Visualizzazione delle definizioni di pipeline

Usa l'interfaccia a riga di comando (CLI) per visualizzare la definizione della pipeline. La CLI stampa un file di definizione della pipeline, in formato JSON. Per ulteriori informazioni sulla sintassi e l'utilizzo di file di definizione della pipeline, vedere [Sintassi del file di definizione della pipeline](#).

Quando si utilizza la CLI, è consigliabile recuperare la definizione della pipeline prima di inviare le modifiche, poiché è possibile che un altro utente o processo abbia modificato la definizione della pipeline dopo l'ultima volta che l'hai utilizzata. Scaricando una copia della definizione corrente e utilizzandola come base per le modifiche, è possibile avere la certezza che si sta utilizzando la definizione di pipeline più recente. È inoltre consigliabile recuperare di nuovo la definizione di pipeline dopo la modifica, in modo da assicurarsi che l'aggiornamento sia riuscito.

Quando usi la CLI, puoi ottenere due diverse versioni della tua pipeline. La versione `active` è la pipeline attualmente in esecuzione. La versione `latest` è una copia creata quando si modifica una pipeline in esecuzione. Quando si carica la pipeline modificata, diventa la versione `active` e quella precedente `active` non è più disponibile.

Per ottenere una definizione di pipeline utilizzando la AWS CLI

Per ottenere la definizione completa della pipeline, utilizzate il [get-pipeline-definition](#) comando. La definizione di pipeline è stampata per l'output standard (stdout).

L'esempio seguente riceve la definizione di pipeline per la pipeline specificata.

```
aws datapipeline get-pipeline-definition --pipeline-id df-00627471S0VYZEXAMPLE
```

Per recuperare una versione specifica di una pipeline, utilizzare l'opzione `--version`. L'esempio seguente recupera la versione `active` della pipeline specificata.

```
aws datapipeline get-pipeline-definition --version active --id df-00627471S0VYZEXAMPLE
```

Visualizzazione dei dettagli dell'istanza della pipeline

È possibile monitorare l'avanzamento della pipeline. Per ulteriori informazioni sullo stato delle istanze, consulta [Interpretazione dei dettagli sullo stato della pipeline](#). Per ulteriori informazioni sulla risoluzione di problemi con istanze della pipeline non eseguite o non completate, consulta [Risoluzione dei problemi più comuni](#).

Per monitorare l'avanzamento di una pipeline tramite la AWS CLI

Per recuperare i dettagli dell'istanza della pipeline, ad esempio uno storico delle volte in cui la pipeline è stata eseguita, utilizzare il comando [list-runs](#). Questo comando consente di filtrare l'elenco di esecuzioni restituite in base al loro stato corrente o agli intervalli di data in cui sono state avviate. Il filtraggio dei risultati è utile perché, a seconda dell'età e della pianificazione della pipeline, la cronologia delle esecuzioni può essere di grandi dimensioni.

L'esempio seguente recupera informazioni per tutte le esecuzioni.

```
aws datapipeline list-runs --pipeline-id df-00627471S0VYZEXAMPLE
```

L'esempio seguente recupera informazioni per tutte le esecuzioni completate.

```
aws datapipeline list-runs --pipeline-id df-00627471S0VYZEXAMPLE --status finished
```

L'esempio seguente recupera informazioni per tutte le esecuzioni avviate nell'intervallo di tempo specificato.

```
aws datapipeline list-runs --pipeline-id df-00627471S0VYZEXAMPLE --start-interval  
"2013-09-02", "2013-09-11"
```

Visualizza log pipeline

La registrazione a livello di pipeline è supportata durante la creazione della pipeline specificando una posizione Amazon S3 nella console o con un `pipelineLogUri` oggetto predefinito in SDK/CLI. La struttura della directory per ogni pipeline all'interno di quella URI è la seguente:

```
pipelineId  
  - componentName  
    - instanceId
```

```
-attemptId
```

Per la pipeline, `df-00123456ABC7DEF8HIJK`, la struttura della directory è simile a:

```
df-00123456ABC7DEF8HIJK
  -ActivityId_fXNzc
    -@ActivityId_fXNzc_2014-05-01T00:00:00
      -@ActivityId_fXNzc_2014-05-01T00:00:00_Attempt=1
```

Per `ShellCommandActivity`, i log per `stderr` e `stdout` associati a queste attività sono memorizzati nella directory per ogni tentativo.

Per le risorse, ad esempio, `EmrCluster`, dove viene impostato un valore `emrLogUri`, tale valore ha la priorità. Altrimenti, le risorse (inclusi `TaskRunner` i registri di tali risorse) seguono la struttura di registrazione della pipeline sopra riportata.

Per visualizzare i log di una determinata pipeline, esegui:

1. Recupera il `ObjectId` chiamando `query-objects` per ottenere l'ID esatto dell'oggetto. Ad esempio:

```
aws datapipeline query-objects --pipeline-id <pipeline-id> --sphere ATTEMPT --region
ap-northeast-1
```

`query-objects` è una CLI impaginata e può restituire un token di impaginazione se ci sono più esecuzioni per quel dato. `pipeline-id` Puoi usare il token per eseguire tutti i tentativi fino a trovare l'oggetto previsto. Ad esempio, un restituito `ObjectId` sarebbe simile a: `@TableBackupActivity_2023-05-02T18:05:18_Attempt=1`.

2. Utilizzando `ObjectId`, recupera la posizione del registro utilizzando:

```
aws datapipeline describe-objects --pipeline-id <pipeline-id> --object-ids <object-id>
--query "pipelineObjects[].fields[?key=='@logLocation'].stringValue"
```

Messaggio di errore di un'attività non riuscita

Per visualizzare il messaggio di errore, verifica innanzitutto l'`ObjectId` utilizzando `query-objects`.

Dopo aver recuperato l'`errorObjectId`, utilizza la `describe-objects` CLI per ottenere il messaggio di errore effettivo.

```
aws datapipeline describe-objects --region ap-northeast-1 --pipeline-id
<pipeline-id> --object-ids <object-id> --query "pipelineObjects[].fields[?
key=='errorMessage'].stringValue"
```

Annullare o rieseguire o contrassegnare come finito un oggetto

Utilizza la `set-status` CLI per annullare un oggetto in esecuzione, eseguire nuovamente un oggetto guasto o contrassegnare un oggetto in esecuzione come Completato.

Innanzitutto, ottieni l'ID dell'oggetto utilizzando la `query-objects` CLI. Ad esempio:

```
aws datapipeline query-objects --pipeline-id <pipeline-id> --sphere INSTANCE --region
ap-northeast-1
```

Usa la `set-status` CLI per modificare lo stato dell'oggetto desiderato. Ad esempio:

```
aws datapipeline set-status --pipeline-id <pipeline-id> --region ap-northeast-1 --status
TRY_CANCEL --object-ids <object-id>
```

Modifica della pipeline

Per modificare un aspetto di una delle pipeline, è possibile aggiornare la definizione di pipeline. Dopo aver modificato una pipeline in esecuzione, è necessario riattivare la pipeline affinché le modifiche diventino effettive. Inoltre, è possibile eseguire nuovamente uno o più componenti della pipeline.

Indice

- [Restrizioni](#)
- [Modifica di una pipeline tramite AWS CLI](#)

Restrizioni

Mentre la pipeline è nello PENDING stato e non è attivata, non è possibile modificarla. Dopo aver attivato una pipeline, è possibile modificare la pipeline con le seguenti limitazioni. Le modifiche apportate si applicano a nuove esecuzioni degli oggetti della pipeline dopo averli salvati, quindi attivare la pipeline di nuovo.

- Non è possibile eliminare un oggetto

- Non è possibile modificare il periodo di pianificazione di un oggetto esistente
- Non è possibile aggiungere, eliminare o modificare campi di riferimento in un oggetto esistente
- Non è possibile fare riferimento a un oggetto esistente in un campo di output di un nuovo oggetto
- Non è possibile modificare la data di inizio pianificata di un oggetto (invece, attivare la pipeline con data e ora specifiche)

Modifica di una pipeline tramite AWS CLI

È possibile modificare una pipeline tramite gli strumenti a riga di comando.

Innanzitutto, scarica una copia della definizione corrente della pipeline utilizzando il [get-pipeline-definition](#) comando. In questo modo, si ha la certezza di modificare la definizione più recente di pipeline. L'esempio seguente stampa la definizione di pipeline in un output standard (stdout).

```
aws datapipeline get-pipeline-definition --pipeline-id df-00627471S0VYZEXAMPLE
```

Salvare la definizione di pipeline su un file e modificarla in base alle esigenze. Aggiorna la definizione della pipeline utilizzando il [put-pipeline-definition](#) comando. L'esempio seguente carica il file di definizione pipeline aggiornato.

```
aws datapipeline put-pipeline-definition --pipeline-id df-00627471S0VYZEXAMPLE --  
pipeline-definition file://MyEmrPipelineDefinition.json
```

È possibile recuperare la definizione di pipeline utilizzando il comando `get-pipeline-definition` per assicurarsi che l'aggiornamento sia stato eseguito correttamente. Per attivare la pipeline, utilizzare il seguente comando [activate-pipeline](#):

```
aws datapipeline activate-pipeline --pipeline-id df-00627471S0VYZEXAMPLE
```

Se si preferisce, attivare la pipeline a partire da una determinata data e ora, utilizzando l'opzione `--start-timestamp` come segue:

```
aws datapipeline activate-pipeline --pipeline-id df-00627471S0VYZEXAMPLE --start-  
timestamp YYYY-MM-DDTHH:MM:SSZ
```

Per eseguire nuovamente uno o più componenti della pipeline, utilizzare il comando [set-status](#).

Clonazione della pipeline

La clonazione crea una copia di una pipeline e consente di specificare un nome per la nuova pipeline. È possibile clonare una pipeline che si trova in qualsiasi stato, anche se presenta errori; tuttavia, la nuova pipeline rimane nello stato di PENDING finché non verrà attivata manualmente. Per la nuova pipeline, l'operazione di clonazione utilizza la versione più recente della definizione originale di pipeline anziché la versione attiva. Nell'operazione di clonazione, la pianificazione completa della pipeline originale non viene copiata nella nuova pipeline, solo l'impostazione del periodo.

Per clonare una pipeline utilizzando la CLI AWS:

1. Crea una nuova pipeline con un nuovo nome e un ID univoco. Annota l'ID della pipeline restituito.
2. Usa la `get-pipeline-definition` CLI per ottenere la definizione della pipeline esistente da clonare e scriverla in un file temporaneo. Nota il percorso assoluto del file.
3. Usa la `put-pipeline-definition` CLI per copiare la definizione della pipeline dalla pipeline esistente alla nuova pipeline.
4. Usa la `get-pipeline-definition` CLI per ottenere la definizione della nuova pipeline e verificare la definizione della pipeline.

```
# Create Pipeline (returns <new-pipeline-id>)
aws datapipeline create-pipeline --name my-cloned-pipeline --unique-id my-cloned-pipeline --region ap-northeast-1

#Get pipeline definition of existing pipeline
aws datapipeline get-pipeline-definition --pipeline-id <existing-pipeline-id> --region ap-northeast-1 > existing_pipeline_definition.json

# Put pipeline definition to new pipeline
aws datapipeline put-pipeline-definition --pipeline-id <new-pipeline-id> --region ap-northeast-1 --pipeline-definition file://<absolute_path_to_existing_pipeline_definition.json>

# get pipeline definition of new pipeline
aws datapipeline get-pipeline-definition --pipeline-id <new-pipeline-id> --region ap-northeast-1
```

Assegnazione di tag alla pipeline

I tag sono coppie chiave-valore che fanno distinzione tra minuscole e maiuscole e contengono una chiave e un valore facoltativo, entrambi definiti dall'utente. È possibile applicare fino a dieci tag a ogni pipeline. Le chiavi del tag devono essere univoche per ciascuna pipeline. Se aggiungi un tag con una chiave già associata alla pipeline, il valore del tag viene aggiornato.

L'applicazione di un tag a una pipeline propaga anche i tag alle risorse sottostanti (ad esempio, cluster Amazon EMR e istanze Amazon EC2). Tuttavia, non applica questi tag alle risorse in uno stato FINISHED o comunque terminato. È possibile utilizzare l'interfaccia a riga di comando per applicare tag a queste risorse, se necessario.

Quando il tag non è più necessario, è possibile eliminarlo dalla pipeline.

Applicare tag alla pipeline utilizzando AWS CLI

Per aggiungere i tag a una nuova pipeline, aggiungere l'opzione `--tags` al comando [create-pipeline](#). Ad esempio, l'opzione seguente crea una pipeline con due tag, un tag `environment` con un valore di `production` e un tag `owner` con un valore di `sales`.

```
--tags key=environment,value=production key=owner,value=sales
```

Per aggiungere i tag a una pipeline esistente, utilizzare il comando [add-tags](#) come segue:

```
aws datapipeline add-tags --pipeline-id df-00627471S0VYZEXAMPLE --tags  
key=environment,value=production key=owner,value=sales
```

Per eliminare i tag da una pipeline esistente, utilizzare il comando [remove-tags](#) come segue:

```
aws datapipeline remove-tags --pipeline-id df-00627471S0VYZEXAMPLE --tag-keys  
environment owner
```

Disattivazione pipeline

La disattivazione di una pipeline in esecuzione sospende la pipeline. Per riprendere l'esecuzione della pipeline, è possibile attivare la pipeline. In questo modo è possibile apportare modifiche. Ad esempio, se si scrivono dati su un database per cui è prevista la manutenzione, è possibile disattivare la pipeline, attenderne il completamento, quindi attivare la pipeline.

Quando si disattiva una pipeline, è possibile specificare il risultato per l'esecuzione di attività. Per impostazione predefinita, queste attività vengono annullate immediatamente. In alternativa, è possibile fare in modo che AWS Data Pipeline attenda il completamento delle attività prima di disattivare la pipeline.

Quando si attiva una pipeline disattivata, è possibile specificare quando riprende. Utilizzando la AWS CLI o l'API, la pipeline riparte dall'ultima esecuzione completata per impostazione predefinita, oppure è possibile specificare la data e l'ora per riprendere la pipeline.

Disattivare la pipeline utilizzando la AWS CLI

Per disattivare la pipeline, utilizza il seguente comando [deactivate-pipeline](#):

```
aws datapipeline deactivate-pipeline --pipeline-id df-00627471SOVYZEXAMPLE
```

Per disattivare la pipeline solo dopo che tutte le attività in esecuzione siano terminate, aggiungere l'opzione `--no-cancel-active` come segue:

```
aws datapipeline deactivate-pipeline --pipeline-id df-00627471SOVYZEXAMPLE --no-cancel-active
```

Quando si è pronti, è possibile riprendere l'esecuzione della pipeline nel punto in cui era stata interrotta utilizzando il seguente comando [attivare-pipeline](#):

```
aws datapipeline activate-pipeline --pipeline-id df-00627471SOVYZEXAMPLE
```

Per avviare la pipeline a partire da una data e ora specifiche, aggiungere l'opzione `--start-timestamp`, come segue:

```
aws datapipeline activate-pipeline --pipeline-id df-00627471SOVYZEXAMPLE --start-timestamp YYYY-MM-DDTHH:MM:SSZ
```

Eliminazione della pipeline

Quando non è più necessario avere una pipeline, ad esempio una pipeline create durante il testing delle applicazioni, è necessario eliminarla per rimuoverla dall'uso attivo. L'eliminazione di una pipeline la inserisce in uno stato di eliminazione. Quando la pipeline è nello stato di eliminazione, la definizione di pipeline e la cronologia delle esecuzioni vengono rimosse. Pertanto, non è più possibile eseguire operazioni sulla pipeline, inclusa la descrizione.

⚠ Important

Non è possibile ripristinare una pipeline eliminata, per cui assicurarsi di non aver bisogno della pipeline in futuro prima di eliminarla.

Per eliminare una pipeline utilizzando la AWS CLI

Per eliminare la pipeline, utilizzare il comando [delete-pipeline](#). Il comando seguente elimina la pipeline specificata.

```
aws datapipeline delete-pipeline --pipeline-id df-00627471S0VYZEXAMPLE
```

Dati e tabelle in gestione temporanea con attività della pipeline

AWS Data Pipeline può eseguire la gestione temporanea dei dati di input e di output nelle pipeline per rendere più facile l'utilizzo di alcune attività, come `ShellCommandActivity` e `HiveActivity`.

La gestione temporanea dei dati consente di copiare i dati dal nodo di dati di input alla risorsa che esegue l'attività e, in modo analogo, dalla risorsa al nodo di dati di output.

I dati staged sulla risorsa Amazon EMR o Amazon EC2 sono disponibili utilizzando variabili speciali nei comandi della shell dell'attività o negli script Hive.

La gestione temporanea delle tabelle è simile a quella dei dati, tranne che i dati in gestione temporanea assumono la forma di tabelle di database, in modo specifico.

AWS Data Pipeline supporta gli scenari seguenti di gestione temporanea:

- Gestione temporanea dei dati con `ShellCommandActivity`
- La gestione temporanea della tabella con Hive e i nodi di dati supportati da tale gestione
- La gestione temporanea della tabella con Hive e i nodi di dati non supportati da tale gestione

ℹ Note

La gestione temporanea funziona solo quando il campo `stage` è impostato su `true` in un'attività, ad esempio `ShellCommandActivity`. Per ulteriori informazioni, consulta [ShellCommandActivity](#).

Inoltre, i nodi di dati e le attività possono essere correlati in quattro modi:

Gestione temporanea dei dati in locale su una risorsa

I dati di input vengono automaticamente copiati nel file system locale della risorsa. I dati di output vengono automaticamente copiati dal file system locale della risorsa al nodo di dati di output. Ad esempio, quando si configurano gli input e gli output `ShellCommandActivity` con gestione temporanea = true, i dati di input sono disponibili come `INPUTx_STAGING_DIR` e i dati di output sono disponibili come `OUTPUTx_STAGING_DIR`, dove x è il numero di input o output.

Gestione temporanea delle definizioni di input e output per un'attività

Il formato di dati di input (nomi delle colonne e nomi delle tabelle) vengono automaticamente copiati nella risorsa dell'attività. Ad esempio, quando si configura `HiveActivity` con gestione temporanea = true. Il formato di dati specificato nell'input `S3DataNode` viene utilizzato per la definizione della tabella dalla tabella Hive.

Gestione temporanea non abilitata

Gli oggetti di input e output e i loro campi sono disponibili per l'attività, mentre i dati no. Ad esempio, `EmrActivity` per impostazione predefinita o quando si configurano altre attività con gestione temporanea = false. In questa configurazione, i campi dati sono disponibili per fare in modo che l'attività effettui un riferimento utilizzando la sintassi dell'espressione AWS Data Pipeline e questo si verifica solo quando la dipendenza viene soddisfatta. Questo serve solo come controllo delle dipendenze. Il codice nell'attività è responsabile della copia dei dati dall'input alla risorsa che esegue l'attività.

Relazione di dipendenza tra gli oggetti

Si è verificato un dipende-dalla relazione tra due oggetti, che comporta una situazione analoga a quando la gestione temporanea non è abilitata. In questo modo, i dati di un nodo o un'attività agiscono come preconditione per l'esecuzione di un'altra attività.

Gestione dei dati con `ShellCommandActivity`

Considerare uno scenario utilizzando un `ShellCommandActivity` con gli oggetti `S3DataNode` come input e output dei dati. AWS Data Pipeline automaticamente gestisce temporaneamente i nodi di dati in modo da renderli accessibili al comando shell come se fossero cartelle di file locali che utilizzano le variabili di ambiente `${INPUT1_STAGING_DIR}` e `${OUTPUT1_STAGING_DIR}` come mostrato nel seguente esempio. La porzione numerica delle variabili denominate

INPUT1_STAGING_DIR e OUTPUT1_STAGING_DIR cresce a seconda del numero di nodi di dati a cui fa riferimento l'attività.

Note

Questo scenario funziona solo se, come descritto, i dati di input e output sono oggetti S3DataNode. Inoltre, la gestione temporanea dei dati di output è consentita solo quando `directoryPath` è impostato sull'oggetto di output S3DataNode.

```
{
  "id": "AggregateFiles",
  "type": "ShellCommandActivity",
  "stage": "true",
  "command": "cat ${INPUT1_STAGING_DIR}/part* > ${OUTPUT1_STAGING_DIR}/aggregated.csv",
  "input": {
    "ref": "MyInputData"
  },
  "output": {
    "ref": "MyOutputData"
  }
},
{
  "id": "MyInputData",
  "type": "S3DataNode",
  "schedule": {
    "ref": "MySchedule"
  },
  "filePath": "s3://my_bucket/source/#{format(@scheduledStartTime, 'YYYY-MM-dd_HH:mm:ss')}/items"
},
{
  "id": "MyOutputData",
  "type": "S3DataNode",
  "schedule": {
    "ref": "MySchedule"
  },
  "directoryPath": "s3://my_bucket/destination/#{format(@scheduledStartTime, 'YYYY-MM-dd_HH:mm:ss')}"
},
```

...

La gestione temporanea della tabella con Hive e i nodi di dati supportati da tale gestione

Considerare uno scenario utilizzando un `HiveActivity` con gli oggetti `S3DataNode` come input e output dei dati. AWS Data Pipeline automaticamente gestisce temporaneamente i nodi di dati in modo da renderli accessibili allo script Hive come se fossero tabelle Hive che utilizzano le variabili `${input1}` e `${output1}` come mostrato nel seguente esempio per `HiveActivity`. La porzione numerica delle variabili denominate `input` e `output` cresce a seconda del numero di nodi di dati a cui fa riferimento l'attività.

Note

Questo scenario funziona solo se, come descritto, i dati di input e output sono oggetti `S3DataNode` o `MySQLDataNode`. La gestione temporanea delle tabelle non è supportata per `DynamoDBDataNode`.

```
{
  "id": "MyHiveActivity",
  "type": "HiveActivity",
  "schedule": {
    "ref": "MySchedule"
  },
  "runsOn": {
    "ref": "MyEmrResource"
  },
  "input": {
    "ref": "MyInputData"
  },
  "output": {
    "ref": "MyOutputData"
  },
  "hiveScript": "INSERT OVERWRITE TABLE ${output1} select * from ${input1};"
},
{
  "id": "MyInputData",
  "type": "S3DataNode",
  "schedule": {
```

```
    "ref": "MySchedule"
  },
  "directoryPath": "s3://test-hive/input"
},
{
  "id": "MyOutputData",
  "type": "S3DataNode",
  "schedule": {
    "ref": "MySchedule"
  },
  "directoryPath": "s3://test-hive/output"
},
...
```

La gestione temporanea della tabella con Hive e i nodi di dati non supportati da tale gestione

Considerare uno scenario utilizzando una `HiveActivity` con `DynamoDBDataNode` come input dei dati e un oggetto `S3DataNode` come output. Non è disponibile alcuna gestione temporanea dei dati `DynamoDBDataNode`, pertanto è necessario innanzitutto creare manualmente la tabella all'interno dello script hive, utilizzando il nome della variabile `#{input.tableName}` per fare riferimento alla tabella DynamoDB. Una nomenclatura simile si applica se la tabella DynamoDB è l'output, tranne che si utilizza una variabile `#{output.tableName}`. La gestione temporanea è disponibile per l'output dell'oggetto `S3DataNode` in questo esempio, pertanto è possibile fare riferimento al nodo dei dati di output come a `#{output1}`.

Note

In questo esempio, la variabile del nome della tabella ha il prefisso di caratteri `#` (hash) perché AWS Data Pipeline utilizza espressioni per accedere a `tableName` o `directoryPath`. Per ulteriori informazioni sul funzionamento della valutazione dell'espressione in AWS Data Pipeline, consulta [Valutazione delle espressioni](#).

```
{
  "id": "MyHiveActivity",
  "type": "HiveActivity",
  "schedule": {
```

```

    "ref": "MySchedule"
  },
  "runsOn": {
    "ref": "MyEmrResource"
  },
  "input": {
    "ref": "MyDynamoData"
  },
  "output": {
    "ref": "MyS3Data"
  },
  "hiveScript": "-- Map DynamoDB Table
SET dynamodb.endpoint=dynamodb.us-east-1.amazonaws.com;
SET dynamodb.throughput.read.percent = 0.5;
CREATE EXTERNAL TABLE dynamodb_table (item map<string,string>)
STORED BY 'org.apache.hadoop.hive.dynamodb.DynamoDBStorageHandler'
TBLPROPERTIES ("dynamodb.table.name" = "#{input.tableName}");
INSERT OVERWRITE TABLE ${output1} SELECT * FROM dynamodb_table;"
},
{
  "id": "MyDynamoData",
  "type": "DynamoDBDataNode",
  "schedule": {
    "ref": "MySchedule"
  },
  "tableName": "MyDDBTable"
},
{
  "id": "MyS3Data",
  "type": "S3DataNode",
  "schedule": {
    "ref": "MySchedule"
  },
  "directoryPath": "s3://test-hive/output"
}
},
...

```

Utilizzo di una pipeline con risorse in più regioni

Per impostazione predefinita, le risorse `Ec2Resource` e `EmrCluster` vengono eseguite nella stessa regione di AWS Data Pipeline; tuttavia, AWS Data Pipeline supporta la possibilità di gestire

i flussi di dati in più regioni, ad esempio le risorse in esecuzione in una regione che consolidano i dati di input da un'altra regione. Consentendo alle risorse di essere eseguite in una determinata regione, si ha anche la flessibilità necessaria per individuare le risorse con i propri dataset dipendenti e per massimizzare le prestazioni riducendo latenze ed evitando costi di trasferimento dei dati in più regioni. È possibile configurare le risorse per l'esecuzione in una regione differente rispetto a AWS Data Pipeline utilizzando il campo `region` in `Ec2Resource` e `EmrCluster`.

Il seguente file JSON della pipeline di esempio mostra come eseguire una `EmrCluster` risorsa nella regione Europa (Irlanda), supponendo che nella stessa regione esista una grande quantità di dati su cui lavorare il cluster. In questo esempio, l'unica differenza rispetto a una tipica pipeline è che `EmrCluster` ha un valore campo `region` impostato su `eu-west-1`.

```
{
  "objects": [
    {
      "id": "Hourly",
      "type": "Schedule",
      "startDateTime": "2014-11-19T07:48:00",
      "endDateTime": "2014-11-21T07:48:00",
      "period": "1 hours"
    },
    {
      "id": "MyCluster",
      "type": "EmrCluster",
      "masterInstanceType": "m3.medium",
      "region": "eu-west-1",
      "schedule": {
        "ref": "Hourly"
      }
    },
    {
      "id": "MyEmrActivity",
      "type": "EmrActivity",
      "schedule": {
        "ref": "Hourly"
      },
      "runsOn": {
        "ref": "MyCluster"
      },
      "step": "/home/hadoop/contrib/streaming/hadoop-streaming.jar, -input, s3n://elasticmapreduce/samples/wordcount/input, -output, s3://eu-west-1-bucket/wordcount/"
    }
  ]
}
```

```
output/#{@scheduledStartTime}, -mapper, s3n://elasticmapreduce/samples/wordcount/  
wordSplitter.py, -reducer, aggregate"  
    }  
  ]  
}
```

La tabella seguente elenca le regioni che è possibile scegliere e i codici di regione associati da utilizzare nel campo `region`.

Note

L'elenco seguente include le regioni in cui è AWS Data Pipeline possibile orchestrare i flussi di lavoro e lanciare risorse Amazon EMR o Amazon EC2. AWS Data Pipeline potrebbe non essere supportato in queste regioni. Per informazioni sulle regioni in cui è supportato AWS Data Pipeline, consulta [Regioni ed endpoint AWS](#).

Nome della regione	Codice regione
Stati Uniti orientali (Virginia settentrionale)	us-east-1
Stati Uniti orientali (Ohio)	us-east-2
Stati Uniti occidentali (California settentrionale)	us-west-1
US West (Oregon)	us-west-2
Canada (Centrale)	ca-central-1
Europa (Irlanda)	eu-west-1
Europe (London)	eu-west-2
Europe (Frankfurt)	eu-central-1
Asia Pacifico (Singapore)	ap-southeast-1
Asia Pacific (Sydney)	ap-southeast-2
Asia Pacifico (Mumbai)	ap-south-1

Nome della regione	Codice regione
Asia Pacifico (Tokyo)	ap-northeast-1
Asia Pacifico (Seoul)	ap-northeast-2
Sud America (São Paulo)	sa-east-1

Guasti di una delle dipendenze e riesecuzioni

AWS Data Pipeline consente di configurare il modo in cui gli oggetti della pipeline si comportano quando una dipendenza si guasta o viene annullata da un utente. È possibile accertarsi che i guasti cadano su altri oggetti della pipeline (consumatori), per evitare attese indefinite. Tutte le attività, i nodi di dati e le precondizioni dispongono di un campo denominato `failureAndRerunMode` con un valore di default di `none`. Per abilitare i guasti di una delle dipendenze, impostare il campo `failureAndRerunMode` su `cascade`.

Quando questo campo è abilitato, i guasti di una delle dipendenze si verificano se l'oggetto di una pipeline è bloccato nello stato `WAITING_ON_DEPENDENCIES` ed eventuali dipendenze hanno dato esito negativo senza alcun comando in sospeso. Durante un guasto di una delle dipendenze, si verificano i seguenti eventi:

- Quando un oggetto ha esito negativo, i suoi consumatori vengono impostati su `CASCADE_FAILED` e sia l'oggetto originale sia le precondizioni dei consumatori vengono impostate su `CANCELED`.
- Tutti gli oggetti già presenti `FINISHED`, `FAILED` oppure `CANCELED` vengono ignorati.

Il guasto di una delle sue dipendenze non funziona sulle dipendenze non riuscite di un oggetto (upstream), tranne per le precondizioni associate all'oggetto originale non riuscito. Gli oggetti di una pipeline interessati da un guasto di una delle dipendenze possono attivare nuovi tentativi o post-operazioni, come, ad esempio `onFail`.

Gli effetti dettagliati di un guasto di una delle dipendenze dipendono dal tipo di oggetto.

Attività

Un'attività cambia in `CASCADE_FAILED` se una delle dipendenze fallisce e, successivamente, innesca un guasto di una delle dipendenze nei consumatori dell'attività. Se una risorsa da cui

dipende un'attività ha esito negativo, l'attività è CANCELED e tutti i suoi consumatori cambiano in CASCADE_FAILED.

Nodi di dati e condizioni preliminari

Se un nodo dati viene configurato come l'output di un'attività che ha esito negativo, il nodo di dati cambia nello stato CASCADE_FAILED. Il guasto di un nodo di dati si propaga a qualsiasi condizione associata che cambia nello stato CANCELED.

Risorse

Se gli oggetti che dipendono da una risorsa sono nello stato FAILED e la risorsa stessa è nello stato WAITING_ON_DEPENDENCIES, allora la risorsa passa allo stato FINISHED.

Riesecuzione di oggetti in cascata con errore

Per impostazione predefinita, rieseguendo qualsiasi attività o nodo di dati si esegue di nuovo solo la risorsa associata. Tuttavia, impostare il campo `failureAndRerunMode` su `cascade` in un oggetto pipeline consente una nuova esecuzione di un comando su un oggetto di destinazione da propagare a tutti i consumatori, nelle seguenti condizioni:

- I consumatori dell'oggetto di destinazione sono nello stato CASCADE_FAILED.
- Le dipendenze dell'oggetto di destinazione non hanno comandi per la riesecuzione in sospeso.
- Le dipendenze dell'oggetto di destinazione non sono nello stato FAILED, CASCADE_FAILED o CANCELED.

Se tenti di eseguire di nuovo un oggetto CASCADE_FAILED e una qualsiasi delle sue dipendenze è FAILED, CASCADE_FAILED oppure CANCELED, la nuova esecuzione fallirà e restituirà l'oggetto allo stato CASCADE_FAILED. Per rieseguire l'oggetto fallito senza errori, è necessario rintracciare l'errore fino alla catena di dipendenza per individuare l'origine dell'errore ed eseguire di nuovo l'oggetto. Quando viene inviato il comando di riesecuzione su una risorsa, si tenta anche di eseguire di nuovo tutti gli oggetti che dipendono da tale risorsa.

Guasto in cascata e riempimenti

Se si abilita il guasto di una delle dipendenze e si dispone di una pipeline che crea molti backfill, gli errori di runtime della pipeline possono comportare la creazione e l'eliminazione di risorse in rapida successione senza eseguire lavoro utile. AWS Data Pipeline tenta di avvisare l'utente

di questa situazione con il seguente messaggio di avvertimento quando si salva una pipeline:
Pipeline_object_name has 'failureAndRerunMode' field set to 'cascade' and you are about to create a backfill with scheduleStartTime *start_time*. This can result in rapid creation of pipeline objects in case of failures. Questo accade perché il guasto di una delle dipendenze è in grado di impostare rapidamente le attività di downstream come CASCADE_FAILED e arrestare i cluster EMR e le risorse EC2 che non sono più necessarie. È consigliabile testare le pipeline con brevi intervalli di tempo per limitare l'impatto di questa situazione.

Sintassi del file di definizione della pipeline

Le istruzioni contenute in questa sezione sono per lavorare manualmente con i file di definizione della pipeline utilizzando l'interfaccia a riga di comando (CLI) di AWS Data Pipeline. Si tratta di un'alternativa per la progettazione interattiva di una pipeline tramite la console AWS Data Pipeline.

È possibile creare manualmente file di definizione della pipeline tramite qualsiasi editor di testo che supporta il salvataggio dei file utilizzando il formato di file UTF-8 e inviare i file utilizzando l'interfaccia a riga di comando di AWS Data Pipeline.

AWS Data Pipeline supporta anche una serie di espressioni complesse e funzioni all'interno di definizioni della pipeline. Per ulteriori informazioni, consulta [Funzioni ed espressioni della pipeline](#).

Struttura dei file

Il primo passo nella creazione della pipeline consiste nel comporre oggetti di definizione della pipeline in un file di definizione della pipeline. L'esempio seguente illustra la struttura generale di un file di definizione della pipeline. Questo file definisce due oggetti, che sono delimitati da "{ and }" e separati da una virgola.

Nell'esempio seguente il primo oggetto definisce due coppie nome-valore, note come campi. Il secondo oggetto definisce tre campi.

```
{
  "objects" : [
    {
      "name1" : "value1",
      "name2" : "value2"
    },
    {
      "name1" : "value3",
```

```
        "name3" : "value4",
        "name4" : "value5"
    }
]
}
```

Quando si crea un file di definizione della pipeline, è necessario selezionare i tipi di oggetti della pipeline di cui hai bisogno, aggiungerli al file di definizione della pipeline, quindi aggiungere i campi appropriati. Per ulteriori informazioni sugli oggetti della pipeline, consulta [Riferimento all'oggetto pipeline](#).

Ad esempio, è possibile creare un oggetto di definizione della pipeline per un nodo di dati di input e un altro per il nodo di dati di output. Quindi crea un altro oggetto di definizione della pipeline per un'attività, ad esempio l'elaborazione dei dati di input utilizzando Amazon EMR.

Campi della pipeline

Dopo aver capito quali tipi di oggetti includere nel file di definizione della pipeline, aggiungere campi alla definizione di ogni oggetto della pipeline. I nomi dei campi vengono inclusi tra virgolette e separati dai valori di campo da uno spazio, una virgola e uno spazio, come mostrato nel seguente esempio.

```
"name" : "value"
```

Il valore del campo può essere una stringa di testo, un riferimento a un altro oggetto, una chiamata di funzione, un'espressione o un elenco ordinato di uno qualsiasi dei tipi precedenti. Per ulteriori informazioni sui tipi di dati che possono essere utilizzati per i valori dei campi, vedi [Tipi di dati di esempio](#). Per ulteriori informazioni sulle funzioni da utilizzare per valutare i valori dei campi, consulta [Valutazione delle espressioni](#).

I campi sono limitati a 2048 caratteri. Gli oggetti possono avere una dimensione pari a 20 KB, il che significa che non è possibile aggiungere molti campi di grandi dimensioni a un oggetto.

Ogni oggetto della pipeline deve contenere i campi riportati di seguito: `id` e `type`, come mostrato nel seguente esempio. Altri campi potrebbero essere richiesti in base al tipo di oggetto. Selezionare un valore `id` significativo per l'utente e univoco all'interno della definizione di pipeline. Il valore per `type` specifica il tipo dell'oggetto. Specificare uno dei tipi di oggetto di definizione della pipeline supportati tra quelli elencati nell'argomento [Riferimento all'oggetto pipeline](#).

```
{
  "id": "MyCopyToS3",
```

```
"type": "CopyActivity"
}
```

Per ulteriori informazioni sui campi obbligatori e facoltativi per ogni oggetto, consulta la documentazione per l'oggetto.

Per includere campi da un oggetto in un altro oggetto, utilizzare il campo `parent` con un riferimento all'oggetto. Ad esempio, l'oggetto "B" include i propri campi, "B1" e "B2", più i campi dell'oggetto "A", "A1" e "A2".

```
{
  "id" : "A",
  "A1" : "value",
  "A2" : "value"
},
{
  "id" : "B",
  "parent" : {"ref" : "A"},
  "B1" : "value",
  "B2" : "value"
}
```

È possibile definire campi comuni in un oggetto con l'ID "Default". Questi campi vengono automaticamente inclusi in ogni oggetto nel file di definizione della pipeline che non imposta esplicitamente il campo `parent` di riferimento a un altro oggetto.

```
{
  "id" : "Default",
  "onFail" : {"ref" : "FailureNotification"},
  "maximumRetries" : "3",
  "workerGroup" : "myWorkerGroup"
}
```

Campi definiti dall'utente

È possibile creare campi personalizzati o definiti dall'utente nei componenti della pipeline e fare riferimento a essi con le espressioni. L'esempio seguente mostra un campo personalizzato denominato `myCustomField` e `my_customFieldReference` aggiunto a un `DataNode` oggetto S3:

```
{
  "id": "S3DataInput",
```

```
"type": "S3DataNode",
"schedule": {"ref": "TheSchedule"},
"filePath": "s3://bucket_name",
"myCustomField": "This is a custom value in a custom field.",
"my_customFieldReference": {"ref": "AnotherPipelineComponent"}
},
```

Un campo definito dall'utente deve avere un nome con prefisso con la parola "my" in tutte lettere minuscole, seguito da una lettera maiuscola o con il carattere di sottolineatura. Inoltre, un campo definito dall'utente può essere un valore di stringa, come l'esempio precedente `myCustomField` o un riferimento a un altro componente della pipeline, come l'esempio precedente `my_customFieldReference`.

Note

Nei campi definiti dall'utente, AWS Data Pipeline controlla solo i riferimenti validi ad altri componenti della pipeline e non i valori di stringhe di campi personalizzati aggiunti dall'utente.

Lavorare con l'API

Note

Se non si è in grado di scrivere con programmi che interagiscono con AWS Data Pipeline, non è necessario installare uno qualsiasi dei kit SDK AWS. Puoi creare ed eseguire pipeline tramite la console o l'interfaccia a riga di comando. Per ulteriori informazioni, consulta [Configurazione di AWS Data Pipeline](#)

Il modo più semplice per scrivere applicazioni che interagiscono con AWS Data Pipeline per implementare un Task Runner personalizzato è di utilizzare uno degli SDK AWS. I kit SDK AWS forniscono funzionalità che semplificano la chiamata delle API del servizio Web dal proprio ambiente di pianificazione preferito. Per ulteriori informazioni, consultare [Installazione del kit SDK AWS](#).

Installazione del kit SDK AWS

Gli SDK AWS forniscono funzioni che eseguono il wrapping dell'API e tengono conto di molti dettagli di connessione, ad esempio il calcolo di firme, la gestione dei tentativi di richiesta e la gestione degli errori. Gli SDK contengono anche codice di esempio, tutorial e altre risorse per aiutarti a

iniziare a scrivere applicazioni che chiamano AWS. La chiamata di funzioni wrapper in un SDK può semplificare notevolmente il processo di scrittura di un'applicazione AWS. Per ulteriori informazioni sul download e l'utilizzo dei kit SDK AWS, consulta [Librerie e codice di esempio](#).

Il supporto AWS Data Pipeline è disponibile in SDK per le seguenti piattaforme:

- [SDK AWS per Java](#)
- [AWS SDK per Node.js](#)
- [SDK AWS per PHP](#)
- [SDK AWS per Python \(Boto\)](#)
- [SDK AWS per Ruby](#)
- [SDK AWS per .NET](#)

Effettuare una richiesta HTTP a AWS Data Pipeline

Per una descrizione completa degli oggetti programmatici in AWS Data Pipeline, consulta la [Documentazione di riferimento delle API di AWS Data Pipeline](#).

Se non utilizzi uno degli SDK AWS, puoi effettuare le operazioni AWS Data Pipeline su HTTP utilizzando il metodo di richiesta POST. Per utilizzare il metodo POST devi specificare l'operazione nell'intestazione della richiesta e fornire i dati per l'operazione in formato JSON nel corpo della richiesta.

Contenuti nell'intestazione HTTP

AWS Data Pipeline richiede le seguenti informazioni nell'intestazione di una richiesta HTTP:

- host Endpoint AWS Data Pipeline.

Per informazioni sugli endpoint consulta [Regioni ed endpoint](#).

- x-amz-date È necessario fornire il timestamp nell'intestazione HTTP Date o nell'intestazione AWS x-amz-date. (Alcune librerie client HTTP non consentono di impostare l'intestazione Date). Quando è presente un'intestazione x-amz-date, il sistema ignora qualsiasi intestazione Date durante l'autenticazione della richiesta.

La data deve essere specificata in uno dei seguenti tre formati, come indicato nel protocollo HTTP/1.1 RFC:

- Dom, 06 novembre 1994 08:49:37 GMT (RFC 822, aggiornato da RFC 1123)

- Domenica, 06-Nov-94 08:49:37 GMT (RFC 850, reso obsoleto da RFC 1036)
- Dom Nov 6 08:49:37 1994 (Formato asctime() ANSI C)
- Authorization Il set di parametri di autorizzazione usato da AWS per garantire la validità e l'autenticità della richiesta. Per ulteriori informazioni su come costruire l'intestazione, consulta [Processo di firma Signature Versione 4](#).
- x-amz-target Il servizio di destinazione della richiesta e l'operazione per i dati, nel formato: <<serviceName>>_<<API version>>.<<operationName>>

Ad esempio, DataPipeline_20121129.ActivatePipeline

- content-type Specifica JSON e la versione. Ad esempio, Content-Type: application/x-amz-json-1.0

Il seguente è un esempio di intestazione per una richiesta HTTP per l'attivazione di una pipeline.

```
POST / HTTP/1.1
host: https://datapipeline.us-east-1.amazonaws.com
x-amz-date: Mon, 12 Nov 2012 17:49:52 GMT
x-amz-target: DataPipeline_20121129.ActivatePipeline
Authorization: AuthParams
Content-Type: application/x-amz-json-1.1
Content-Length: 39
Connection: Keep-Alive
```

Contenuto del corpo HTTP

Il corpo di una richiesta HTTP contiene i dati per l'operazione specificata nell'intestazione di una richiesta HTTP. I dati devono essere formattati in base allo schema dei dati JSON per ogni API AWS Data Pipeline. Lo schema di dati JSON AWS Data Pipeline definisce i tipi di dati e i parametri (ad esempio operatori di confronto e costanti di enumerazione) disponibili per ciascuna operazione.

Formattare il corpo di una richiesta HTTP

Utilizza il formato dati JSON per trasmettere simultaneamente i valori dei dati e la struttura corrispondente. Gli elementi possono essere annidati all'interno di altri elementi utilizzando la notazione parentesi. L'esempio seguente mostra una richiesta per mettere una definizione di pipeline composta da tre oggetti e relativi slot.

```
{
  "pipelineId": "df-00627471S0VYZEXAMPLE",
  "pipelineObjects":
  [
    {"id": "Default",
     "name": "Default",
     "slots":
     [
       {"key": "workerGroup",
        "stringValue": "MyWorkerGroup"}
     ]
    },
    {"id": "Schedule",
     "name": "Schedule",
     "slots":
     [
       {"key": "startDateTime",
        "stringValue": "2012-09-25T17:00:00"},
       {"key": "type",
        "stringValue": "Schedule"},
       {"key": "period",
        "stringValue": "1 hour"},
       {"key": "endDateTime",
        "stringValue": "2012-09-25T18:00:00"}
     ]
    },
    {"id": "SayHello",
     "name": "SayHello",
     "slots":
     [
       {"key": "type",
        "stringValue": "ShellCommandActivity"},
       {"key": "command",
        "stringValue": "echo hello"},
       {"key": "parent",
        "refValue": "Default"},
       {"key": "schedule",
        "refValue": "Schedule"}
     ]
    }
  ]
}
```

Gestire la risposta HTTP

Si elencano di seguito alcune intestazioni importanti nella risposta HTTP e il modo in cui vanno gestite nell'applicazione:

- **HTTP/1.1** Questa intestazione è seguita da un codice di stato. Un valore del codice di 200 indica un'operazione riuscita. Qualsiasi altro valore indica un errore.
- **x-amzn-RequestId** Questa intestazione contiene un ID della richiesta che è possibile utilizzare per risolvere i problemi relativi a una richiesta AWS Data Pipeline. L'esempio dell'ID di una richiesta è K2QH8DNOU907N97FNA2GDLL8OBVV4KQNSO5AEMVJF66Q9ASUAAJG.
- **x-amz-crc32**—AWS Data Pipeline calcola un checksum CRC32 del payload HTTP e lo restituisce nell'intestazione **x-amz-crc32**. Consigliamo di calcolare il checksum CRC32 lato client e di confrontarlo con l'intestazione **x-amz-crc32**; se i checksum non corrispondono, è possibile che i dati sono stati danneggiati in transito. In questo caso, è necessario effettuare di nuovo la richiesta.

Gli utenti di SDK AWS non devono eseguire manualmente questa verifica, perché gli SDK calcolano il checksum di ogni risposta da Amazon DynamoDB e automaticamente riprovano nel caso in cui venga rilevata una discrepanza.

Richiesta e risposta AWS Data Pipeline JSON di esempio

I seguenti esempi mostrano una richiesta per la creazione di una nuova pipeline. Quindi viene mostrata la risposta AWS Data Pipeline, incluso l'identificatore della pipeline appena creata.

Richiesta HTTP POST

```
POST / HTTP/1.1
host: https://datapipeline.us-east-1.amazonaws.com
x-amz-date: Mon, 12 Nov 2012 17:49:52 GMT
x-amz-target: DataPipeline_20121129.CreatePipeline
Authorization: AuthParams
Content-Type: application/x-amz-json-1.1
Content-Length: 50
Connection: Keep-Alive

{"name": "MyPipeline",
 "uniqueId": "12345ABCDEF"}
```

Risposta AWS Data Pipeline

```
HTTP/1.1 200
x-amzn-RequestId: b16911ce-0774-11e2-af6f-6bc7a6be60d9
x-amz-crc32: 2215946753
Content-Type: application/x-amz-json-1.0
Content-Length: 2
Date: Mon, 16 Jan 2012 17:50:53 GMT

{"pipelineId": "df-00627471S0VYZEXAMPLE"}
```

Sicurezza in AWS Data Pipeline

Per AWS, la sicurezza del cloud ha la massima priorità. In quanto cliente AWS, puoi trarre vantaggio da un'architettura di data center e di rete progettata per soddisfare i requisiti delle aziende più esigenti a livello di sicurezza.

La sicurezza è una responsabilità condivisa tra te e AWS. Il modello di [responsabilità condivisa](#) descrive questo modello come sicurezza del cloud e sicurezza nel cloud:

- La sicurezza del cloud: AWS è responsabile della protezione dell'infrastruttura che esegue i servizi AWS nel AWS Cloud. AWS fornisce, inoltre, servizi utilizzabili in modo sicuro. I revisori di terze parti testano regolarmente e verificano l'efficacia della nostra sicurezza nell'ambito dei [Programmi di conformità AWS](#). Per informazioni sui programmi di conformità applicabili a AWS Data Pipeline, consulta [Servizi AWS coperti dal programma di compliance](#).
- Sicurezza nel cloud: la tua responsabilità è determinata dal servizio AWS che viene utilizzato. L'utente è anche responsabile per altri fattori, tra cui la riservatezza dei dati, i requisiti dell'azienda e leggi e normative applicabili.

Questa documentazione consente di comprendere come applicare il modello di responsabilità condivisa quando si usa AWS Data Pipeline. I seguenti argomenti illustrano come configurare AWS Data Pipeline per soddisfare gli obiettivi di sicurezza e conformità. Vengono inoltre fornite informazioni su come utilizzare altri servizi AWS che consentono di monitorare e proteggere le risorse AWS Data Pipeline.

Argomenti

- [Protezione dei dati in AWS Data Pipeline](#)
- [Identity and Access Management per AWS Data Pipeline](#)
- [Logging e monitoraggio in AWS Data Pipeline](#)
- [Risposta agli eventi imprevisti in AWS Data Pipeline](#)
- [Convalida della conformità per AWS Data Pipeline](#)
- [Resilienza in AWS Data Pipeline](#)
- [Sicurezza dell'infrastruttura in AWS Data Pipeline](#)
- [Analisi della configurazione e delle vulnerabilità in AWS Data Pipeline](#)

Protezione dei dati in AWS Data Pipeline

Il [modello di responsabilità condivisa](#) di AWS si applica alla protezione dei dati in AWS Data Pipeline. Come descritto in questo modello, AWS è responsabile della protezione dell'infrastruttura globale che esegue tutto l'Cloud AWS. L'utente è responsabile di mantenere il controllo sui contenuti ospitati su questa infrastruttura. Questi contenuti comprendono la configurazione della protezione e le 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.

Per garantire la protezione dei dati, ti suggeriamo di proteggere le credenziali Account AWS e di configurare i singoli utenti con AWS IAM Identity Center o AWS Identity and Access Management (IAM). In questo modo, a ogni utente verranno assegnate solo le autorizzazioni necessarie per svolgere il suo lavoro. Ti suggeriamo, inoltre, di proteggere i dati nei seguenti modi:

- Utilizza l'autenticazione a più fattori (MFA) con ogni account.
- Utilizza SSL/TLS per comunicare con le risorse AWS. È consigliabile TLS 1.2 o versioni successive.
- Configura la registrazione delle API e delle attività degli utenti con AWS CloudTrail.
- Utilizza le soluzioni di crittografia AWS, insieme a tutti i controlli di sicurezza di default all'interno dei Servizi AWS.
- Utilizza i servizi di sicurezza gestiti avanzati, ad esempio Amazon Macie, che aiutano a individuare e proteggere i dati sensibili archiviati in Amazon S3.
- Se si richiedono moduli crittografici convalidati FIPS 140-2 quando si accede ad AWS tramite una CLI o un'API, utilizzare un endpoint FIPS. Per ulteriori informazioni sugli endpoint FIPS disponibili, consulta il [Federal Information Processing Standard \(FIPS\) 140-2](#).
- AWS Data Pipeline supporta IMDSv2 per le risorse Amazon EMR e Amazon EC2. Per utilizzare IMDSv2 con Amazon EMR, 5.27.1, 5.27.1 e 6.2 o successive. Per ulteriori informazioni, consulta [Configurare le richieste di servizi di metadati per le istanze Amazon EC2](#) e [Utilizzare IMDSv2](#).

Ti consigliamo vivamente di non inserire mai informazioni identificative sensibili, ad esempio gli indirizzi e-mail dei clienti, nei tag o nei campi di testo in formato libero, ad esempio un campo Name (Nome). Questo include il lavoro con AWS Data Pipeline o altri Servizi AWS utilizzando la console, l'API, la AWS CLI o gli SDK AWS. I dati inseriti nei tag o nei campi di testo in formato libero utilizzati per i nomi possono essere utilizzati per i log di fatturazione o 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.

Identity and Access Management per AWS Data Pipeline

Le credenziali di sicurezza identificano l'utente per i servizi su AWS e concedono le autorizzazioni per utilizzare le risorse AWS, ad esempio le pipeline. Puoi utilizzare le caratteristiche di AWS Data Pipeline e AWS Identity and Access Management (IAM) per consentire a AWS Data Pipeline ad altri utenti di accedere alle risorse AWS Data Pipeline senza condividere le credenziali di sicurezza.

Le organizzazioni possono condividere l'accesso alle pipeline, in modo che i singoli utenti in quella organizzazione siano in grado di svilupparle e mantenerle in modo collaborativo. Tuttavia, per esempio, può essere necessario eseguire le operazioni seguenti:

- Controlla quali utenti possono accedere a pipeline specifiche
- Proteggere una pipeline di produzione per evitare che venga modificata per errore
- Consentire a entità di controllo l'accesso in sola lettura alle pipeline, ma impedire loro di apportare modifiche

AWS Data Pipeline è integrato con AWS Identity and Access Management (IAM), che offre una vasta gamma di funzionalità:

- Creare utenti e gruppi nel tuo Account AWS.
- Condividi facilmente AWS le tue risorse tra gli utenti del tuo Account AWS.
- Assegnare credenziali di sicurezza univoche a ciascun utente.
- Controllare l'accesso di ciascun utente a servizi e risorse.
- Ottieni una fattura unica per tutti gli utenti del tuo Account AWS.

Utilizzando IAM con AWS Data Pipeline, puoi controllare se gli utenti all'interno dell'organizzazione sono in grado di eseguire un'attività utilizzando specifiche operazioni API e se possono utilizzare risorse AWS specifiche. Puoi utilizzare le policy IAM basate sui tag delle pipeline e sui gruppi di lavoro per condividere le tue pipeline con altri utenti e controllare il loro livello di accesso.

Indice

- [Policy IAM per AWS Data Pipeline](#)

- [Esempi di policy per AWS Data Pipeline](#)
- [Ruoli IAM per AWS Data Pipeline](#)

Policy IAM per AWS Data Pipeline

Per impostazione predefinita predefinita, le entità IAM non dispongono dell'autorizzazione per creare o modificare risorse AWS. Per permettere alle entità IAM di creare o modificare le risorse ed eseguire le attività, devi creare policy IAM che concedano alle entità IAM l'autorizzazione a utilizzare specifiche risorse e operazioni API e quindi collegare tali policy alle entità IAM che richiedono tali autorizzazioni.

Quando si collega una policy a un utente o a un gruppo di utenti, tramite essa viene concessa o rifiutata agli utenti l'autorizzazione per l'esecuzione delle attività specificate per le risorse specificate. Per informazioni generali sulle policy IAM, consulta la sezione relativa ad [autorizzazioni e policy](#) nella Guida per l'utente di IAM. Per ulteriori informazioni sulla gestione e la creazione di policy IAM personalizzate, consulta la sezione relativa alla [gestione delle policy IAM](#).

Indice

- [Sintassi delle policy](#)
- [Controllo degli accessi alle pipeline tramite i tag](#)
- [Controllo degli accessi alle pipeline tramite i gruppi di lavoratori](#)

Sintassi delle policy

Una policy IAM è un documento JSON costituito da una o più dichiarazioni. Ogni dichiarazione è strutturata come segue:

```
{
  "Statement": [{
    "Effect": "effect",
    "Action": "action",
    "Resource": "*",
    "Condition": {
      "condition": {
        "key": "value"
      }
    }
  ]
}
```

```
}  
]  
}
```

Una dichiarazione di policy include i seguenti elementi:

- **Effect (Effetto):** l'elemento effect può essere Allow o Deny. Per impostazione predefinita predefinita predefinita predefinita predefinita predefinita predefinita predefinita predefinita predefinita, le entità IAM non dispongono dell'autorizzazione per l'utilizzo di risorse e operazioni API, pertanto tutte le richieste vengono rifiutate. Un permesso esplicito sostituisce l'impostazione predefinita. Un rifiuto esplicito sovrascrive tutti i consensi.
- **Action (Operazione):** l'elemento action corrisponde all'operazione API specifica per la quale si concede o si nega l'autorizzazione. Per un elenco di azioni perAWS Data Pipeline, consulta [Azioni](#) nel riferimentoAWS Data Pipeline API.
- **Resource (Risorsa):** la risorsa che viene modificata dall'operazione. L'unico valore valido qui è "*".
- **Condition:** le condizioni sono facoltative. Possono essere utilizzate per controllare quando sarà in vigore una policy.

AWS Data Pipeline implementa le chiavi di contesto AWS (consulta la sezione relativa alle [chiavi disponibili](#)), oltre alle seguenti chiavi specifiche per i servizi.

- `datapipeline:PipelineCreator`— Concedano l'accesso all'utente che ha creato la pipeline. Per un esempio, vedi [Concedere al proprietario della pipeline l'accesso completo](#).
- `datapipeline:Tag`— Concedere l'accesso in base all'etichettatura della pipeline. Per ulteriori informazioni, consulta [Controllo degli accessi alle pipeline tramite i tag](#).
- `datapipeline:workerGroup`— Concedano l'accesso in base al nome del gruppo di lavoratori. Per ulteriori informazioni, consulta [Controllo degli accessi alle pipeline tramite i gruppi di lavoratori](#).

Controllo degli accessi alle pipeline tramite i tag

Puoi creare criteri IAM che fanno riferimento ai tag per la pipeline. In questo modo è possibile utilizzare l'applicazione di tag alle pipeline per eseguire quanto segue:

- Concedere l'accesso in sola lettura a una pipeline
- Concedere l'accesso in lettura/scrittura a una pipeline
- Bloccare l'accesso a una pipeline

Ad esempio, supponiamo che un responsabile disponga di due ambienti di pipeline, produzione e sviluppo, e di un gruppo IAM per ogni ambiente. Per le pipeline nell'ambiente di produzione, il manager concede l'accesso di lettura/scrittura agli utenti del gruppo IAM di produzione, ma concede l'accesso in sola lettura agli utenti del gruppo IAM di sviluppo. Per le pipeline nell'ambiente di sviluppo, il manager concede l'accesso di lettura/scrittura sia ai gruppi IAM di produzione che a quelli di sviluppo.

Per realizzare questo scenario, il manager etichetta le pipeline di produzione con il tag «environment=production» e allega la seguente politica al gruppo IAM di sviluppatori. La prima istruzione concede l'accesso in sola lettura a tutte le pipeline. La seconda istruzione consente l'accesso in lettura/scrittura alle pipeline che non dispongono di un tag "environment=production".

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "datapipeline:Describe*",
        "datapipeline:ListPipelines",
        "datapipeline:GetPipelineDefinition",
        "datapipeline:QueryObjects"
      ],
      "Resource": "*"
    },
    {
      "Effect": "Allow",
      "Action": "datapipeline:*",
      "Resource": "*",
      "Condition": {
        "StringNotEquals": {"datapipeline:Tag/environment": "production"}
      }
    }
  ]
}
```

Inoltre, il manager allega la seguente politica al gruppo IAM di produzione. Questa istruzione concede l'accesso completo a tutte le pipeline.

```
{
  "Version": "2012-10-17",
```

```
"Statement": [  
  {  
    "Effect": "Allow",  
    "Action": "datapipeline:*",  
    "Resource": "*"  
  }  
]  
}
```

Per ulteriori esempi, vedi [Concessione agli utenti dell'accesso in sola lettura basato su un tag](#) e [Concessione agli utenti dell'accesso completo basato su un tag](#).

Controllo degli accessi alle pipeline tramite i gruppi di lavoratori

È possibile creare politiche IAM che fanno riferimento ai nomi dei gruppi di lavoro.

Ad esempio, supponiamo che un responsabile disponga di due ambienti di pipeline, produzione e sviluppo, e di un gruppo IAM per ogni ambiente. Il responsabile ha tre server di database con runner di attività configurati, rispettivamente, per gli ambienti di produzione, pre-produzione e sviluppo. Il manager desidera garantire che gli utenti del gruppo IAM di produzione possano creare pipeline che trasferiscono le attività alle risorse di produzione e che gli utenti del gruppo IAM di sviluppo possano creare pipeline che trasferiscono le attività alle risorse di preproduzione e di sviluppo.

Per ottenere questo scenario, il responsabile installa runner di attività sulle risorse di produzione con le credenziali relative e imposta `workerGroup` su "prodresource". Inoltre, il responsabile installa il runner delle attività sulle risorse di sviluppo con le credenziali di sviluppo e imposta `workerGroup` su "pre-produzione" e "sviluppo". Il manager allega la seguente politica al gruppo IAM degli sviluppatori per bloccare l'accesso alle risorse «prodresource». La prima istruzione concede l'accesso in sola lettura a tutte le pipeline. La seconda istruzione consente di concedere l'accesso in lettura/scrittura alle pipeline quando il nome del gruppo di lavoratori ha il prefisso "dev" o "pre-prod".

```
{  
  "Version": "2012-10-17",  
  "Statement": [  
    {  
      "Effect": "Allow",  
      "Action": [  
        "datapipeline:Describe*",  
        "datapipeline:ListPipelines",  
        "datapipeline:GetPipelineDefinition",  
        "datapipeline:QueryObjects"  
      ]  
    }  
  ]  
}
```

```

    ],
    "Resource": "*"
  },
  {
    "Action": "datapipeline:*",
    "Effect": "Allow",
    "Resource": "*",
    "Condition": {
      "StringLike": {
        "datapipeline:workerGroup": ["dev*", "pre-prod*"]
      }
    }
  }
]
}

```

Inoltre, il manager allega la seguente politica al gruppo IAM di produzione per concedere l'accesso alle risorse «prodresource». La prima istruzione concede l'accesso in sola lettura a tutte le pipeline. La seconda istruzione consente l'accesso in lettura/scrittura quando il nome del gruppo di lavoratori ha il prefisso "prod".

```

{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "datapipeline:Describe*",
        "datapipeline:ListPipelines",
        "datapipeline:GetPipelineDefinition",
        "datapipeline:QueryObjects"
      ],
      "Resource": "*"
    },
    {
      "Effect": "Allow",
      "Action": "datapipeline:*",
      "Resource": "*",
      "Condition": {
        "StringLike": {"datapipeline:workerGroup": "prodresource*"}
      }
    }
  ]
}

```

```
}
```

Esempi di policy per AWS Data Pipeline

I seguenti esempi illustrano come concedere agli utenti accesso completo o limitato alle pipeline.

Indice

- [Esempio 1: Concedere agli utenti accesso in sola lettura in base a un tag](#)
- [Esempio 2: Concedere agli utenti l'accesso completo in base a un tag](#)
- [Esempio 3: Concedere al proprietario della pipeline l'accesso completo](#)
- [Esempio 4: concedere agli utenti l'accesso alla console AWS Data Pipeline](#)

Esempio 1: Concedere agli utenti accesso in sola lettura in base a un tag

La policy seguente consente agli utenti di usare le operazioni dell'API AWS Data Pipeline di sola lettura, ma solo con le pipeline che hanno il tag "environment=production".

L'operazione ListPipelines API non supporta entrambi l'autorizzazione basata su tag.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "datapipeline:Describe*",
        "datapipeline:GetPipelineDefinition",
        "datapipeline:ValidatePipelineDefinition",
        "datapipeline:QueryObjects"
      ],
      "Resource": [
        "*"
      ],
      "Condition": {
        "StringEquals": {
          "datapipeline:Tag/environment": "production"
        }
      }
    }
  ]
}
```

```
]
}
```

Esempio 2: Concedere agli utenti l'accesso completo in base a un tag

La seguente politica consente agli utenti di utilizzare tutte le azioni AWS Data Pipeline API, ad eccezione ListPipelines, ma solo, delle pipeline che hanno il tag «environment=test».

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "datapipeline:*"
      ],
      "Resource": [
        "*"
      ],
      "Condition": {
        "StringEquals": {
          "datapipeline:Tag/environment": "test"
        }
      }
    }
  ]
}
```

Esempio 3: Concedere al proprietario della pipeline l'accesso completo

La policy seguente consente agli utenti di usare tutte le operazioni dell'API AWS Data Pipeline, ma solo con le proprie pipeline.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "datapipeline:*"
      ],
      "Resource": [
```

```

        "*"
    ],
    "Condition": {
        "StringEquals": {
            "datapipeline:PipelineCreator": "${aws:userid}"
        }
    }
}
]
}

```

Esempio 4: concedere agli utenti l'accesso alla console AWS Data Pipeline

La policy seguente consente agli utenti di creare e gestire una pipeline utilizzando la console AWS Data Pipeline.

Questa policy include l'operazione per le autorizzazioni `PassRole` per le risorse specifiche vincolate al `roleARN` che AWS Data Pipeline richiede. Per ulteriori informazioni sull'`PassRole` autorizzazione basata sull'identità (IAM), consulta il post del blog [Concedere l'autorizzazione al lancio di istanze EC2 con ruoli IAM \(PassRole autorizzazione\)](#).

```

{
  "Version": "2012-10-17",
  "Statement": [{
    "Action": [
      "cloudwatch:*",
      "datapipeline:*",
      "dynamodb:DescribeTable",
      "elasticmapreduce:AddJobFlowSteps",
      "elasticmapreduce:ListInstance*",
      "iam:AddRoleToInstanceProfile",
      "iam:CreateInstanceProfile",
      "iam:GetInstanceProfile",
      "iam:GetRole",
      "iam:GetRolePolicy",
      "iam:ListInstanceProfiles",
      "iam:ListInstanceProfilesForRole",
      "iam:ListRoles",
      "rds:DescribeDBInstances",
      "rds:DescribeDBSecurityGroups",
      "redshift:DescribeClusters",
      "redshift:DescribeClusterSecurityGroups",
      "s3:List*",
    ]
  }
]
}

```

```
    "sns:ListTopics"
  ],
  "Effect": "Allow",
  "Resource": [
    "*"
  ]
},
{
  "Action": "iam:PassRole",
  "Effect": "Allow",
  "Resource": [
    "arn:aws:iam::*:role/DataPipelineDefaultResourceRole",
    "arn:aws:iam::*:role/DataPipelineDefaultRole"
  ]
}
]
```

Ruoli IAM per AWS Data Pipeline

AWS Data Pipeline utilizza iAWS Identity and Access Management ruoli. Le politiche di autorizzazione associate ai ruoli IAM determinano quali azioniAWS Data Pipeline e le applicazioni possono eseguire e a qualiAWS risorse possono accedere. Per ulteriori informazioni, consulta [Ruoli IAM](#) nella Guida per l'utente IAM.

AWS Data Pipeline richiede due ruoli IAM:

- Il ruolo della pipeline controllaAWS Data Pipeline l'accesso alle risorse di AWS. Nelle definizioni degli oggetti della pipeline, il `role` campo specifica questo ruolo.
- Il ruolo dell'istanza EC2 controlla l'accesso delle applicazioni in esecuzione sulle istanze EC2, comprese le istanze EC2 nei cluster Amazon EMR, alleAWS risorse. Nelle definizioni degli oggetti della pipeline, il `resourceRole` campo specifica questo ruolo.

Important

Se hai creato una pipeline prima del 3 ottobre 2022 utilizzando laAWS Data Pipeline console con ruoli predefiniti, l'haiAWS Data Pipeline creato `DataPipelineDefaultRole` per te e hai collegato la policy `AWSDataPipelineRole` gestita al ruolo. A partire dal 3 ottobre 2022, la policy `AWSDataPipelineRole` gestita è obsoleta e il ruolo della pipeline deve essere specificato per una pipeline quando si utilizza la console.

Ti consigliamo di esaminare le pipeline esistenti e determinare se `DataPipelineDefaultRole` sono associate alla pipeline e se `AWSDataPipelineRole` sono collegate a quel ruolo. In tal caso, verifica l'accesso consentito da questa politica per assicurarti che sia appropriato per i tuoi requisiti di sicurezza. Aggiungi, aggiorna o sostituisci le politiche e le dichiarazioni politiche allegate a questo ruolo, se necessario. In alternativa, puoi aggiornare una pipeline per utilizzare un ruolo creato con politiche di autorizzazione diverse.

Esempi di politiche di autorizzazione per AWS Data Pipeline i ruoli

A ogni ruolo sono associate una o più politiche di autorizzazione che determinano le AWS risorse a cui il ruolo può accedere e le azioni che il ruolo può eseguire. Questo argomento fornisce un esempio di politica di autorizzazione per il ruolo della pipeline. Fornisce inoltre il contenuto di `AmazonEC2RoleforDataPipelineRole`, che è la politica gestita per il ruolo di istanza EC2 predefinito, `DataPipelineDefaultResourceRole`.

Esempio di policy di autorizzazione del ruolo di pipeline

La politica di esempio che segue ha lo scopo di consentire le funzioni essenziali che AWS Data Pipeline richiedono l'esecuzione di una pipeline con risorse Amazon EC2 e Amazon EMR. Fornisce inoltre autorizzazioni per l'accesso ad altre AWS risorse, come Amazon S3 e Amazon S3, che richiedono molte pipeline. Se gli oggetti definiti in una pipeline non richiedono le risorse di un AWS servizio, si consiglia vivamente di rimuovere le autorizzazioni per accedere a tale servizio. Ad esempio, se la pipeline non definisce [DynamoDB DataNode](#) o non utilizza l'[SnsAlarm](#) azione, si consiglia di rimuovere le istruzioni di autorizzazione per tali azioni.

- Sostituisci `111122223333` con il tuo ID account AWS.
- Sostituisci `NameOfDataPipelineRole` con il nome del ruolo della pipeline (il ruolo a cui è associata questa politica).
- Sostituisci `NameOfDataPipelineResourceRole` con il nome del ruolo dell'istanza EC2.
- Sostituisci `us-west-1` con la regione appropriata per la tua applicazione.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
```

```
"Effect": "Allow",
"Action": [
    "iam:GetInstanceProfile",
    "iam:GetRole",
    "iam:GetRolePolicy",
    "iam:ListAttachedRolePolicies",
    "iam:ListRolePolicies",
    "iam:PassRole"
],
"Resource": [
    "arn:aws:iam::111122223333:role/NameOfDataPipelineRole",
    "arn:aws:iam::111122223333 :role/NameOfDataPipelineResourceRole"
]
},
{
    "Effect": "Allow",
    "Action": [
        "ec2:AuthorizeSecurityGroupEgress",
        "ec2:AuthorizeSecurityGroupIngress",
        "ec2:CancelSpotInstanceRequests",
        "ec2:CreateNetworkInterface",
        "ec2:CreateSecurityGroup",
        "ec2:CreateTags",
        "ec2>DeleteNetworkInterface",
        "ec2>DeleteSecurityGroup",
        "ec2>DeleteTags",
        "ec2:DescribeAvailabilityZones",
        "ec2:DescribeAccountAttributes",
        "ec2:DescribeDhcpOptions",
        "ec2:DescribeImages",
        "ec2:DescribeInstanceStatus",
        "ec2:DescribeInstances",
        "ec2:DescribeKeyPairs",
        "ec2:DescribeLaunchTemplates",
        "ec2:DescribeNetworkAcls",
        "ec2:DescribeNetworkInterfaces",
        "ec2:DescribePrefixLists",
        "ec2:DescribeRouteTables",
        "ec2:DescribeSecurityGroups",
        "ec2:DescribeSpotInstanceRequests",
        "ec2:DescribeSpotPriceHistory",
        "ec2:DescribeSubnets",
        "ec2:DescribeTags",
        "ec2:DescribeVpcAttribute",
```

```

        "ec2:DescribeVpcEndpoints",
        "ec2:DescribeVpcEndpointServices",
        "ec2:DescribeVpcs",
        "ec2:DetachNetworkInterface",
        "ec2:ModifyImageAttribute",
        "ec2:ModifyInstanceAttribute",
        "ec2:RequestSpotInstances",
        "ec2:RevokeSecurityGroupEgress",
        "ec2:RunInstances",
        "ec2:TerminateInstances",
        "ec2:DescribeVolumeStatus",
        "ec2:DescribeVolumes",
        "elasticmapreduce:TerminateJobFlows",
        "elasticmapreduce:ListSteps",
        "elasticmapreduce:ListClusters",
        "elasticmapreduce:RunJobFlow",
        "elasticmapreduce:DescribeCluster",
        "elasticmapreduce:AddTags",
        "elasticmapreduce:RemoveTags",
        "elasticmapreduce:ListInstanceGroups",
        "elasticmapreduce:ModifyInstanceGroups",
        "elasticmapreduce:GetCluster",
        "elasticmapreduce:DescribeStep",
        "elasticmapreduce:AddJobFlowSteps",
        "elasticmapreduce:ListInstances",
        "iam:ListInstanceProfiles",
        "redshift:DescribeClusters"
    ],
    "Resource": [
        "*"
    ]
},
{
    "Effect": "Allow",
    "Action": [
        "sns:GetTopicAttributes",
        "sns:Publish"
    ],
    "Resource": [
        "arn:aws:sns:us-west-1:111122223333:MyFirstSNSTopic",
        "arn:aws:sns:us-west-1:111122223333:MySecondSNSTopic",
        "arn:aws:sns:us-west-1:111122223333:AnotherSNSTopic"
    ]
},

```

```
{
  "Effect": "Allow",
  "Action": [
    "s3:ListBucket",
    "s3:ListMultipartUploads"
  ],
  "Resource": [
    "arn:aws:s3:::MyStagingS3Bucket",
    "arn:aws:s3:::MyLogsS3Bucket",
    "arn:aws:s3:::MyInputS3Bucket",
    "arn:aws:s3:::MyOutputS3Bucket",
    "arn:aws:s3:::AnotherRequiredS3Buckets"
  ]
},
{
  "Effect": "Allow",
  "Action": [
    "s3:GetObject",
    "s3:GetObjectMetadata",
    "s3:PutObject"
  ],
  "Resource": [
    "arn:aws:s3:::MyStagingS3Bucket/*",
    "arn:aws:s3:::MyLogsS3Bucket/*",
    "arn:aws:s3:::MyInputS3Bucket/*",
    "arn:aws:s3:::MyOutputS3Bucket/*",
    "arn:aws:s3:::AnotherRequiredS3Buckets/*"
  ]
},
{
  "Effect": "Allow",
  "Action": [
    "dynamodb:Scan",
    "dynamodb:DescribeTable"
  ],
  "Resource": [
    "arn:aws:dynamodb:us-west-1:111122223333:table/MyFirstDynamoDBTable",
    "arn:aws:dynamodb:us-west-1:111122223333:table/MySecondDynamoDBTable",
    "arn:aws:dynamodb:us-west-1:111122223333:table/AnotherDynamoDBTable"
  ]
},
{
  "Effect": "Allow",
  "Action": [
```

```

        "rds:DescribeDBInstances"
    ],
    "Resource": [
        "arn:aws:rds:us-west-1:111122223333:db:MyFirstRdsDb",
        "arn:aws:rds:us-west-1:111122223333:db:MySecondRdsDb",
        "arn:aws:rds:us-west-1:111122223333:db:AnotherRdsDb"
    ]
}
]
}

```

Policy gestita predefinita per il ruolo dell'istanza EC2

Il contenuto di `AmazonEC2RoleforDataPipelineRole` è mostrato di seguito.

Questa è la politica gestita associata al ruolo di risorsa predefinito per AWS Data Pipeline, `DataPipelineDefaultResourceRole`. Quando si definisce un ruolo di risorsa per la pipeline, si consiglia di iniziare con questa politica di autorizzazione e quindi rimuovere le autorizzazioni per le azioni di AWS servizio che non sono necessarie.

Viene mostrata la versione 3 della politica, che è la versione più recente al momento della stesura di questo documento. Visualizza la versione più recente della policy utilizzando la console IAM.

```

{
  "Version": "2012-10-17",
  "Statement": [{
    "Effect": "Allow",
    "Action": [
      "cloudwatch:*",
      "datapipeline:*",
      "dynamodb:*",
      "ec2:Describe*",
      "elasticmapreduce:AddJobFlowSteps",
      "elasticmapreduce:Describe*",
      "elasticmapreduce:ListInstance*",
      "elasticmapreduce:ModifyInstanceGroups",
      "rds:Describe*",
      "redshift:DescribeClusters",
      "redshift:DescribeClusterSecurityGroups",
      "s3:*",
      "sdb:*",
      "sns:*",
      "sqs:*"
    ]
  }]
}

```

```
    ],  
    "Resource": ["*"]  
  }  
}
```

Creazione di ruoli IAM AWS Data Pipeline e modifica delle autorizzazioni dei ruoli

Utilizza le procedure seguenti per creare ruoli per l'AWS Data Pipeline utilizzando la console IAM. Il processo è costituito da due fasi. Per prima cosa devi creare una policy di autorizzazione da collegare al ruolo. Successivamente, devi creare il ruolo e collega quindi la policy. Dopo aver creato un ruolo, puoi modificarne le autorizzazioni allegando e scollegando i criteri di autorizzazione.

Note

Quando si creano ruoli per l'AWS Data Pipeline utilizzando la console come descritto di seguito, IAM crea e allega le politiche di fiducia appropriate richieste dal ruolo.

Per creare una politica di autorizzazioni da utilizzare con un ruolo per AWS Data Pipeline

1. Apri la console IAM all'indirizzo <https://console.aws.amazon.com/iam/>.
2. Nel riquadro di navigazione, selezionare Policies e scegliere Create Policy (Crea policy).
3. Scegliere la scheda JSON.
4. Se state creando un ruolo di pipeline, copiate e incollate il contenuto dell'esempio di policy e modificalo in [Esempio di policy di autorizzazione del ruolo di pipeline](#) base ai vostri requisiti di sicurezza. In alternativa, se state creando un ruolo di istanza EC2 personalizzato, fate lo stesso per l'esempio in [Policy gestita predefinita per il ruolo dell'istanza EC2](#).
5. Scegli Review policy (Esamina policy).
6. Inserisci un nome per la politica, ad esempio, MyDataPipelineRolePolicy e una descrizione facoltativa, quindi scegli Crea politica.
7. Prendere nota del nome della policy. Questa informazione è necessaria in fase di creazione del ruolo.

Per creare un ruolo IAM per AWS Data Pipeline

1. Apri la console IAM all'indirizzo <https://console.aws.amazon.com/iam/>.

2. Nel riquadro di navigazione, scegliere Roles (Ruoli) e quindi Create role (Crea ruolo).
3. In Scegli un caso d'uso, scegli Data Pipeline.
4. In Seleziona il tuo caso d'uso, esegui una delle seguenti operazioni:
 - SceglieteData Pipeline di creare un ruolo nella pipeline.
 - ScegliEC2 Role for Data Pipeline di creare un ruolo di risorsa.
5. Scegliere Successivo: Autorizzazioni.
6. SeAWS Data Pipeline è elencata la politica predefinita per, procedi con i seguenti passaggi per creare il ruolo, quindi modificalo in base alle istruzioni della procedura successiva. In caso contrario, inserisci il nome della policy creata nella procedura precedente, quindi selezionala dall'elenco.
7. Scegli Avanti: Tag, inserisci i tag da aggiungere al ruolo, quindi scegli Avanti: Revisione.
8. Inserisci un nome per il ruolo, ad esempioMyDataPipelineRole, e una descrizione facoltativa, quindi scegli Crea ruolo.

Per allegare o scollegare una politica di autorizzazioni per un ruolo IAM perAWS Data Pipeline

1. Aprire la console IAM all'indirizzo <https://console.aws.amazon.com/iam/>.
2. Nel riquadro di navigazione, selezionare Roles (Ruoli).
3. Nella casella di ricerca, inizia a digitare il nome del ruolo che desideri modificare, ad esempio DataPipelineDefaultRoleo MyDataPipelineRole, quindi scegli il nome del ruolo dall'elenco.
4. Nella scheda Autorizzazioni, segui questi passaggi:
 - Per scollegare un criterio di autorizzazione, in Criteri di autorizzazione, scegli il pulsante di rimozione all'estrema destra della voce del criterio. Scegli Detach quando ti viene richiesto di confermare.
 - Per allegare un criterio creato in precedenza, scegli Allega criteri. Nella casella di ricerca, inizia a digitare il nome della policy che desideri modificare, selezionala dall'elenco, quindi scegli Collegare policy.

Modifica dei ruoli per una pipeline esistente

Se desideri assegnare un ruolo di pipeline o un ruolo di risorsa diverso a una pipeline, puoi utilizzare l'editor di architettura nellaAWS Data Pipeline console.

Per modificare i ruoli assegnati a una pipeline utilizzando la console

1. Apri laAWS Data Pipeline console all'[indirizzo https://console.aws.amazon.com/datapipeline/](https://console.aws.amazon.com/datapipeline/).
2. Selezionate la pipeline dall'elenco, quindi scegliete Azioni, Modifica.
3. Nel riquadro destro dell'editor dell'architetto, scegli Altri.
4. Dagli elenchi Ruolo e ruolo della risorsa, scegli i ruoliAWS Data Pipeline che desideri assegnare, quindi scegli Salva.

Logging e monitoraggio in AWS Data Pipeline

AWS Data Pipelineè integrato conAWS CloudTrail, un servizio che offre un record delle operazioni eseguite da un utente, un ruolo o unAWS servizio inAWS Data Pipeline. CloudTrail acquisisce tutte le chiamate APIAWS Data Pipeline come eventi. Le chiamate acquisite includono le chiamate dalla console di AWS Data Pipeline e le chiamate di codice alle operazioni delle API AWS Data Pipeline. Se si crea un percorso, è possibile abilitare la distribuzione continua di CloudTrail eventi in un bucket Amazon S3, inclusi gli eventi perAWS Data Pipeline. Se non configuri un percorso, è comunque possibile visualizzare gli eventi più recenti nella CloudTrail console di in Cronologia eventi. Le informazioni raccolte da CloudTrail, consentono di determinare la richiesta effettuata aAWS Data Pipeline, l'indirizzo IP di origine da cui è stata eseguita la richiesta, l'autore della richiesta, il momento in cui è stata eseguita e altri dettagli.

Per ulteriori informazioni CloudTrail, consulta la [Guida perAWS CloudTrail l'utente](#).

AWS Data PipelineInformazioni in CloudTrail

CloudTrail è abilitato sull'AWSaccount al momento della sua creazione. Quando si verifica un'attività inAWS Data Pipeline, questa viene registrata in un CloudTrail evento insieme ad altri eventiAWS di servizio nella cronologia eventi. È possibile visualizzare, cercare e scaricare gli eventi recenti nell'account AWS. Per ulteriori informazioni, consulta [Visualizzazione di eventi mediante la cronologia CloudTrail eventi](#) di.

Per una registrazione continua degli eventi nell'account AWS che includa gli eventi per AWS Data Pipeline, creare un trail. Un trail consente di CloudTrail distribuire i file di log in un bucket Amazon 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 registro nel bucket Amazon S3 specificato. Inoltre, puoi configurare altriAWS servizi per analizzare

con maggiore dettaglio e usare i dati raccolti nei CloudTrail log. Per ulteriori informazioni, consulta gli argomenti seguenti:

- [Panoramica della creazione di un percorso](#)
- [CloudTrail Servizi e integrazioni supportati](#)
- [Configurazione delle Amazon SNS per CloudTrail](#)
- [Ricezione di file di CloudTrail log da più regioni](#) e [Ricezione di file di CloudTrail log da più account](#)

Tutte le AWS Data Pipeline azioni vengono registrate CloudTrail e documentate nel [capitolo AWS Data Pipeline API Reference Actions](#). Ad esempio, le chiamate alle CreatePipelineoperazioni generano voci nei file di CloudTrail log.

Ogni evento o voce del log contiene informazioni sull'utente che ha generato la richiesta. Le informazioni di identità consentono di determinare quanto segue:

- Se la richiesta è stata effettuata con credenziali di ruolo IAM o root.
- Se la richiesta è stata effettuata con le credenziali di sicurezza temporanee per un ruolo o un utente federato.
- Se la richiesta è stata effettuata da un altro servizio AWS.

Per ulteriori informazioni, consulta [Elemento CloudTrail userIdentity](#).

Comprensione delle voci dei file di log di AWS Data Pipeline

Un trail è una configurazione che consente la distribuzione di eventi come i file di log in un bucket Amazon S3 specificato dall'utente. CloudTrail i file di log possono contenere una o più voci di log. Un evento rappresenta una singola richiesta da un'origine e include informazioni sull'operazione richiesta, data e ora dell'operazione, parametri della richiesta e così via. CloudTrail i file di log non sono una traccia di stack delle chiamate API pubbliche e di conseguenza non devono apparire in base a un ordine specifico.

L'esempio seguente mostra una voce di CloudTrail log di che illustra l>CreatePipelineoperazione:

```
{
  "Records": [
    {
```

```
"eventVersion": "1.02",
"userIdentity": {
  "type": "Root",
  "principalId": "123456789012",
  "arn": "arn:aws:iam::aws-account-id:role/role-name",
  "accountId": "role-account-id",
  "accessKeyId": "role-access-key"
},
"eventTime": "2014-11-13T19:15:15Z",
"eventSource": "datapipeline.amazonaws.com",
"eventName": "CreatePipeline",
"awsRegion": "us-east-1",
"sourceIPAddress": "72.21.196.64",
"userAgent": "aws-cli/1.5.2 Python/2.7.5 Darwin/13.4.0",
"requestParameters": {
  "name": "testpipeline",
  "uniqueId": "sounique"
},
"responseElements": {
  "pipelineId": "df-06372391ZG65EXAMPLE"
},
"requestID": "65cbf1e8-6b69-11e4-8816-cfcbadd04c45",
"eventID": "9f99dce0-0864-49a0-bffa-f72287197758",
"eventType": "AwsApiCall",
"recipientAccountId": "role-account-id"
},
...additional entries
]
}
```

Risposta agli eventi imprevisti in AWS Data Pipeline

La risposta agli eventi imprevisti per AWS Data Pipeline è una responsabilità AWS. AWS dispone di una policy e un programma formali e documentati che regolano la risposta agli eventi imprevisti.

I problemi operativi AWS con ampio impatto sono pubblicati nel (Pannello di controllo dello stato dei servizi) AWS Service Health Dashboard . Problemi operativi sono anche pubblicati su singoli account tramite il Personal Health Dashboard.

Convalida della conformità per AWS Data Pipeline

AWS Data Pipeline non rientra nell'ambito dei programmi di conformità di AWS. Per un elenco di servizi AWS nell'ambito di programmi di conformità specifici, consulta [Servizi AWS coperti dal programma di compliance](#). Per informazioni, consulta [Programmi per la conformità di AWS](#).

Resilienza in AWS Data Pipeline

L'infrastruttura globale di AWS è basata su regioni e zone di disponibilità AWS. Le regioni forniscono più zone di disponibilità fisicamente separate e isolate che sono connesse tramite reti altamente ridondanti, a bassa latenza e velocità effettiva elevata. Con le zone di disponibilità, è possibile progettare e gestire applicazioni e database che eseguono il failover automatico 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.

Per ulteriori informazioni sulle regioni AWS e sulle zone di disponibilità, consulta [Infrastruttura globale di AWS](#).

Sicurezza dell'infrastruttura in AWS Data Pipeline

Come servizio gestito, AWS Data Pipeline è protetto dalle procedure di sicurezza di rete globali di AWS descritte nel [Amazon Web Services: Panoramica sui processi di sicurezza di](#) [whitepaper](#).

Utilizza le chiamate API pubblicate di AWS per accedere a AWS Data Pipeline tramite la rete. I client devono supportare Transport Layer Security (TLS) 1.0 o versioni successive. È consigliabile TLS 1.2 o versioni successive. I client devono, inoltre, supportare le suite di cifratura con PFS (Perfect Forward Secrecy), 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. In alternativa, è possibile utilizzare [AWS Security Token Service](#) (AWS STS) per generare le credenziali di sicurezza temporanee per sottoscrivere le richieste.

Analisi della configurazione e delle vulnerabilità in AWS Data Pipeline

Configurazione e controllo IT sono una responsabilità condivisa tra AWS e te, il nostro cliente. Per ulteriori informazioni, consulta il [modello di responsabilità condivisa di AWS](#).

Tutorial

I tutorial seguenti illustrano il step-by-step processo di creazione e utilizzo delle pipeline con. AWS Data Pipeline

Tutorial

- [Elabora i dati utilizzando Amazon EMR con Hadoop Streaming](#)
- [Copia dati CSV tra bucket Amazon S3 utilizzando AWS Data Pipeline](#)
- [Esporta dati MySQL su Amazon S3 utilizzando AWS Data Pipeline](#)
- [Copia i dati su Amazon Redshift utilizzando AWS Data Pipeline](#)

Elabora i dati utilizzando Amazon EMR con Hadoop Streaming

Puoi utilizzarlo AWS Data Pipeline per gestire i tuoi cluster Amazon EMR. AWS Data PipelinePuoi specificare i prerequisiti che devono essere soddisfatti prima del lancio del cluster (ad esempio, assicurando che i dati odierni siano caricati su Amazon S3), una pianificazione per l'esecuzione ripetuta del cluster e la configurazione del cluster da utilizzare. Il seguente tutorial ti guiderà attraverso l'avvio di un semplice cluster.

In questo tutorial, crei una pipeline per un semplice cluster Amazon EMR per eseguire un job Hadoop Streaming preesistente fornito da Amazon EMR e inviare una notifica Amazon SNS dopo il completamento dell'attività con successo. Per questa attività, utilizzi la risorsa cluster Amazon EMR fornita AWS Data Pipeline da. L'applicazione di esempio si chiama WordCount e può anche essere eseguita manualmente dalla console Amazon EMR. Tieni presente che i cluster generati da per tuo AWS Data Pipeline conto vengono visualizzati nella console Amazon EMR e fatturati sul tuo account AWS.

Oggetti della pipeline

La pipeline utilizza i seguenti oggetti:

[EmrActivity](#)

Definisce il lavoro da eseguire nella pipeline (esegue un processo Hadoop Streaming preesistente fornito da Amazon EMR).

[EmrCluster](#)

La risorsa che AWS Data Pipeline usa per eseguire questa attività.

Un cluster è un insieme di istanze Amazon EC2. AWS Data Pipeline avvia il cluster e lo termina al termine dell'attività.

[Pianificazione](#)

Data di avvio, ora e durata di questa attività. È anche possibile specificare la data e l'ora di fine.

[SnsAlarm](#)

Invia una notifica Amazon SNS all'argomento specificato dopo il completamento dell'attività con successo.

Indice

- [Prima di iniziare](#)
- [Avvio di un cluster mediante la riga di comando](#)

Prima di iniziare

Assicurarsi di aver effettuato le operazioni seguenti.

- Completare le operazioni descritte in [Configurazione di AWS Data Pipeline](#).
- (facoltativo) Imposta un VPC per il cluster e un gruppo di sicurezza per il VPC.
- Creare un argomento per l'invio di e-mail di notifica e annotare l'ARN (Amazon Resource Name) dell'argomento. Per ulteriori informazioni, consultare la pagina relativa alla [Creazione di un argomento](#) in Nozioni di base del servizio di notifiche di Amazon Simple.

Avvio di un cluster mediante la riga di comando

Se gestisci regolarmente un cluster Amazon EMR per analizzare i log web o eseguire analisi di dati scientifici, puoi utilizzarlo AWS Data Pipeline per gestire i tuoi cluster Amazon EMR. Con AWS Data Pipeline, puoi specificare i prerequisiti che devono essere soddisfatti prima del lancio del cluster (ad esempio, assicurandoti che i dati odierni siano caricati su Amazon S3). Questo tutorial illustra il lancio di un cluster che può essere un modello per una semplice pipeline basata su Amazon EMR o come parte di una pipeline più complessa.

Prerequisiti

Prima di poter usare la CLI, devi completare le fasi seguenti:

1. Installa e configura un'interfaccia a riga di comando (CLI). Per ulteriori informazioni, consulta [Accesso a AWS Data Pipeline](#).
2. Assicurati che i ruoli IAM siano denominati `DataPipelineDefaultRole` e `DataPipelineDefaultResourceRole` esistano. La AWS Data Pipeline console crea questi ruoli automaticamente. Se non hai usato la AWS Data Pipeline console almeno una volta, devi creare questi ruoli manualmente. Per ulteriori informazioni, consulta [Ruoli IAM per AWS Data Pipeline](#).

Processi

- [Creazione del file di definizione della pipeline](#)
- [Caricamento e attivazione della definizione della pipeline](#)
- [Monitorare le esecuzioni della pipeline](#)

Creazione del file di definizione della pipeline

Il codice seguente è il file di definizione della pipeline per un semplice cluster Amazon EMR che esegue un job di streaming Hadoop esistente fornito da Amazon EMR. Questa applicazione di esempio si chiama `WordCount` e puoi anche eseguirla utilizzando la console Amazon EMR.

Copiare questo codice in un file di testo e salvarlo come `MyEmrPipelineDefinition.json`. Dovresti sostituire la posizione del bucket Amazon S3 con il nome di un bucket Amazon S3 di tua proprietà. È inoltre necessario sostituire le date di inizio e fine. Per avviare immediatamente i cluster, impostare `startTime` su una data di un giorno nel passato e `endTime` su un giorno nel futuro. AWS Data Pipeline quindi inizia lanciando i cluster "scaduti" immediatamente nel tentativo di gestire ciò che viene percepito come un backlog di lavoro. Questo backfill significa che non è necessario attendere un'ora per vedere AWS Data Pipeline lanciare il primo cluster.

```
{
  "objects": [
    {
      "id": "Hourly",
      "type": "Schedule",
      "startTime": "2012-11-19T07:48:00",
      "endTime": "2012-11-21T07:48:00",
```

```
    "period": "1 hours"
  },
  {
    "id": "MyCluster",
    "type": "EmrCluster",
    "masterInstanceType": "m1.small",
    "schedule": {
      "ref": "Hourly"
    }
  },
  {
    "id": "MyEmrActivity",
    "type": "EmrActivity",
    "schedule": {
      "ref": "Hourly"
    },
    "runsOn": {
      "ref": "MyCluster"
    },
    "step": "/home/hadoop/contrib/streaming/hadoop-streaming.jar, -input, s3n://elasticmapreduce/samples/wordcount/input, -output, s3://myawsbucket/wordcount/output/#{@scheduledStartTime}, -mapper, s3n://elasticmapreduce/samples/wordcount/wordSplitter.py, -reducer, aggregate"
  }
]
}
```

Questa pipeline ha tre oggetti:

- `Hourly`, che rappresenta la pianificazione del lavoro. È possibile impostare una pianificazione come uno dei campi di un'attività. Quando si effettua questa operazione, l'attività viene eseguita in base alla pianificazione, oppure in questo caso, su base oraria.
- `MyCluster`, che rappresenta il set di istanze Amazon EC2 utilizzate per eseguire il cluster. È possibile specificare il numero e la dimensione delle istanze EC2 da eseguire come cluster. Se non si specifica il numero di istanze, il cluster ne usa due, un nodo master e un nodo di task. È possibile specificare una sottorete in cui avviare il cluster. Puoi aggiungere configurazioni aggiuntive al cluster, ad esempio azioni di bootstrap per caricare software aggiuntivo sull'AMI fornita da Amazon EMR.
- `MyEmrActivity`, che rappresenta il calcolo da elaborare con il cluster. Amazon EMR supporta diversi tipi di cluster, tra cui streaming, Cascading e Scripted Hive. Il `runsOn` campo fa riferimento a `MyCluster`, utilizzandolo come specifica per le basi del cluster.

Caricamento e attivazione della definizione della pipeline

Devi caricare la definizione della pipeline e attivare la pipeline. Nei seguenti comandi di esempio, sostituisci *pipeline_name* con un'etichetta per la tua pipeline e *pipeline_file con il percorso completo per il file di definizione della pipeline*.json

AWS CLI

Per creare la definizione della pipeline e attivare la pipeline, usa il seguente comando [create-pipeline](#). Nota l'ID della tua pipeline, perché utilizzerai questo valore con la maggior parte dei comandi CLI.

```
aws datapipeline create-pipeline --name pipeline_name --unique-id token
{
  "pipelineId": "df-00627471S0VYZEXAMPLE"
}
```

Per caricare la definizione della pipeline, usa il seguente [put-pipeline-definition](#) comando.

```
aws datapipeline put-pipeline-definition --pipeline-id df-00627471S0VYZEXAMPLE --
pipeline-definition file://MyEmrPipelineDefinition.json
```

Se la pipeline viene convalidata correttamente, il `validationErrors` campo è vuoto. È necessario esaminare eventuali avvertenze.

Per attivare la pipeline, usa il seguente comando [activate-pipeline](#).

```
aws datapipeline activate-pipeline --pipeline-id df-00627471S0VYZEXAMPLE
```

Puoi verificare che la tua pipeline appaia nell'elenco delle pipeline utilizzando il seguente comando [list-pipelines](#).

```
aws datapipeline list-pipelines
```

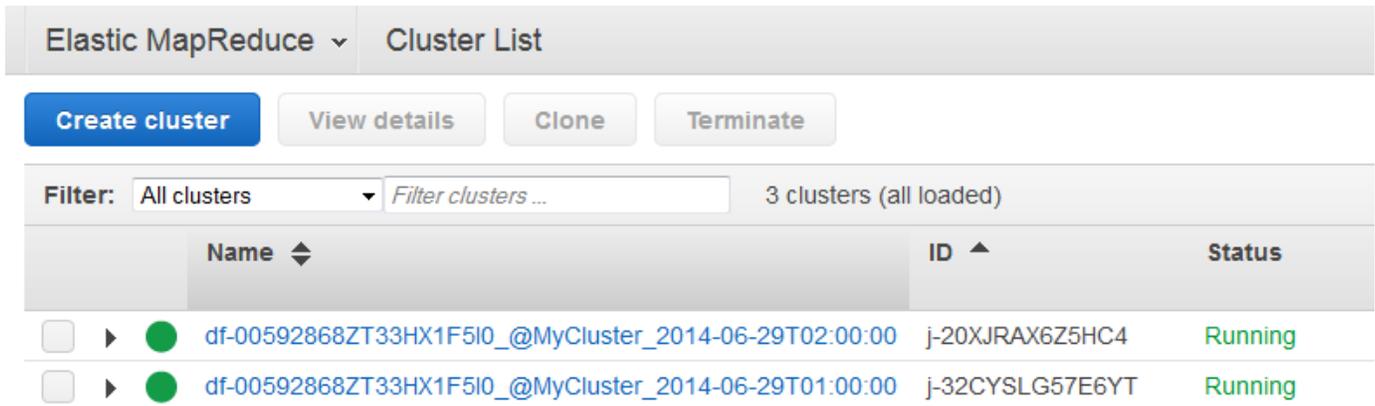
Monitorare le esecuzioni della pipeline

Puoi visualizzare i cluster avviati AWS Data Pipeline utilizzando la console Amazon EMR e puoi visualizzare la cartella di output utilizzando la console Amazon S3.

Per controllare lo stato di avanzamento di cluster avviate da AWS Data Pipeline

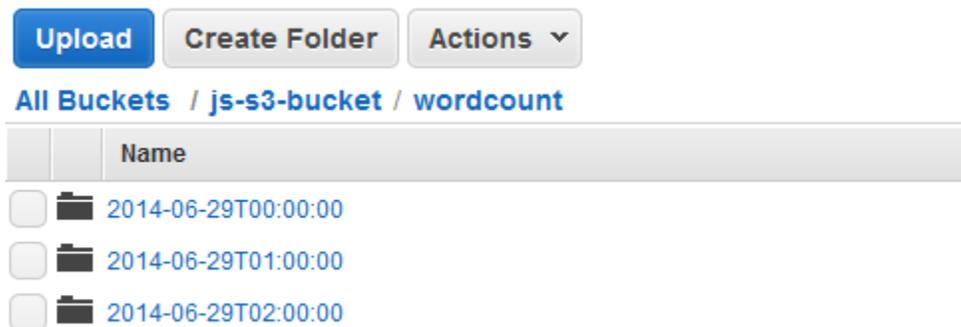
1. Apri la console Amazon EMR.

2. I cluster generati da AWS Data Pipeline hanno un nome formattato come segue: `_@ < > _ . emr-cluster-name <pipeline-identifier><launch-time>`



Name	ID	Status
df-00592868ZT33HX1F5I0_@MyCluster_2014-06-29T02:00:00	j-20XJRAX6Z5HC4	Running
df-00592868ZT33HX1F5I0_@MyCluster_2014-06-29T01:00:00	j-32CYSLG57E6YT	Running

3. Al termine di una delle esecuzioni, apri la console Amazon S3 e verifica che la cartella di output con timestamp esista e contenga i risultati previsti del cluster.



Name
2014-06-29T00:00:00
2014-06-29T01:00:00
2014-06-29T02:00:00

Copia dati CSV tra bucket Amazon S3 utilizzando AWS Data Pipeline

Dopo aver letto [Cos'è AWS Data Pipeline?](#) e deciso come utilizzare AWS Data Pipeline per automatizzare il trasferimento e la trasformazione dei dati, è ora di iniziare a utilizzare la creazione delle pipeline di dati. Per aiutare a comprendere come funziona AWS Data Pipeline, esaminiamo un'attività semplice.

Questo tutorial illustra il processo di creazione di una pipeline di dati per copiare dati da un bucket Amazon S3 a un altro e quindi inviare una notifica Amazon SNS dopo il completamento dell'attività di copia. Utilizzerai un'istanza EC2 gestita da AWS Data Pipeline per questa attività di copia.

Oggetti della pipeline

La pipeline utilizza i seguenti oggetti:

[CopyActivity](#)

L'attività AWS Data Pipeline eseguita per questa pipeline (copia dei dati CSV da un bucket Amazon S3 a un altro).

Important

Esistono dei limiti quando si usa il formato di file CSV con CopyActivity e S3DataNode. Per ulteriori informazioni, consulta [CopyActivity](#).

[Pianificazione](#)

Data di avvio, ora e ricorrenza di questa attività. È anche possibile specificare la data e l'ora di fine.

[Ec2Resource](#)

La risorsa (un'istanza EC2) che AWS Data Pipeline utilizza per eseguire questa attività.

[S3 DataNode](#)

I nodi di input e output (bucket Amazon S3) per questa pipeline.

[SnsAlarm](#)

L'azione AWS Data Pipeline deve essere eseguita quando vengono soddisfatte le condizioni specificate (inviare notifiche Amazon SNS a un argomento dopo il completamento dell'attività con successo).

Indice

- [Prima di iniziare](#)
- [Copia dei dati CSV tramite la riga di comando](#)

Prima di iniziare

Assicurarsi di aver effettuato le operazioni seguenti.

- Completare le operazioni descritte in [Configurazione di AWS Data Pipeline](#).
- (facoltativo) Impostare un VPC per l'istanza e un gruppo di sicurezza per il VPC.
- Crea un bucket Amazon S3 come origine dati.

Per ulteriori informazioni, consulta [Create a Bucket \(Creazione di un bucket\)](#) nella Guida per l'utente di Amazon Simple Storage Service.

- Carica i tuoi dati nel tuo bucket Amazon S3.

Per ulteriori informazioni, consulta [Aggiunta di un oggetto a un bucket](#) nella Guida per l'utente di Amazon Simple Storage Service.

- Crea un altro bucket Amazon S3 come destinazione dati
- Creare un argomento per l'invio di e-mail di notifica e annotare l'ARN (Amazon Resource Name) dell'argomento. Per ulteriori informazioni, consultare la pagina relativa alla [Creazione di un argomento](#) in Nozioni di base del servizio di notifiche di Amazon Simple.
- (Facoltativo) Questo tutorial utilizza le policy del ruolo IAM di default create da AWS Data Pipeline. Se preferisci creare e configurare la tua politica di ruolo IAM e le tue relazioni di fiducia, segui le istruzioni descritte in [Ruoli IAM per AWS Data Pipeline](#).

Copia dei dati CSV tramite la riga di comando

Puoi creare e utilizzare pipeline per copiare dati da un bucket Amazon S3 a un altro.

Prerequisiti

Prima di iniziare , devi completare le fasi seguenti:

1. Installa e configura un'interfaccia a riga di comando (CLI). Per ulteriori informazioni, consulta [Accesso a AWS Data Pipeline](#).
2. Assicurati che i ruoli IAM siano denominati DataPipelineDefaultRoleed DataPipelineDefaultResourceRoleesistano. La AWS Data Pipeline console crea questi ruoli automaticamente. Se non hai usato la AWS Data Pipeline console almeno una volta, devi creare questi ruoli manualmente. Per ulteriori informazioni, consulta [Ruoli IAM per AWS Data Pipeline](#).

Processi

- [Definire una pipeline in formato JSON](#)
- [Caricamento e attivazione della definizione della pipeline](#)

Definire una pipeline in formato JSON

Questo scenario di esempio mostra come utilizzare le definizioni della pipeline JSON e l'interfaccia a AWS Data Pipeline riga di comando per pianificare la copia dei dati tra due bucket Amazon S3 in un intervallo di tempo specifico. Questo è il file JSON completo di definizione della pipeline seguito da una spiegazione per ciascuna delle sue sezioni.

Note

È consigliabile utilizzare un editor di testo che può aiutare a verificare la sintassi di file in formato JSON e nominare il file utilizzando l'estensione del file `.json`.

In questo esempio, per chiarezza, abbiamo saltato i campi facoltativi e abbiamo mostrato solo i campi obbligatori. Il file JSON completo della pipeline per questo esempio è:

```
{
  "objects": [
    {
      "id": "MySchedule",
      "type": "Schedule",
      "startDateTime": "2013-08-18T00:00:00",
      "endDateTime": "2013-08-19T00:00:00",
      "period": "1 day"
    },
    {
      "id": "S3Input",
      "type": "S3DataNode",
      "schedule": {
        "ref": "MySchedule"
      },
      "filePath": "s3://example-bucket/source/inputfile.csv"
    },
    {
      "id": "S3Output",
      "type": "S3DataNode",
      "schedule": {
        "ref": "MySchedule"
      },
      "filePath": "s3://example-bucket/destination/outputfile.csv"
    }
  ]
}
```

```
    "id": "MyEC2Resource",
    "type": "Ec2Resource",
    "schedule": {
      "ref": "MySchedule"
    },
    "instanceType": "m1.medium",
    "role": "DataPipelineDefaultRole",
    "resourceRole": "DataPipelineDefaultResourceRole"
  },
  {
    "id": "MyCopyActivity",
    "type": "CopyActivity",
    "runsOn": {
      "ref": "MyEC2Resource"
    },
    "input": {
      "ref": "S3Input"
    },
    "output": {
      "ref": "S3Output"
    },
    "schedule": {
      "ref": "MySchedule"
    }
  }
]
}
```

Pianificazione

La pipeline definisce una pianificazione con un data di inizio e di fine, insieme a un periodo per stabilire con quale frequenza viene eseguita l'attività in questa pipeline.

```
{
  "id": "MySchedule",
  "type": "Schedule",
  "startDateTime": "2013-08-18T00:00:00",
  "endDateTime": "2013-08-19T00:00:00",
  "period": "1 day"
},
```

Nodi dati Amazon S3

Successivamente, il componente della DataNode pipeline S3 di input definisce una posizione per i file di input; in questo caso, una posizione del bucket Amazon S3. Il DataNode componente S3 di input è definito dai seguenti campi:

```
{
  "id": "S3Input",
  "type": "S3DataNode",
  "schedule": {
    "ref": "MySchedule"
  },
  "filePath": "s3://example-bucket/source/inputfile.csv"
},
```

Id

Il nome definito dall'utente per il percorso di input (un'etichetta solo di riferimento).

Type (Tipo)

Il tipo di componente della pipeline, che è «DataNodeS3» in base alla posizione in cui risiedono i dati, in un bucket Amazon S3.

Pianificazione

Un riferimento al componente di pianificazione che abbiamo creato nelle righe precedenti del file JSON denominato «». MySchedule

Percorso

Il percorso ai dati associati al nodo di dati. La sintassi per un nodo di dati è determinata dal tipo. Ad esempio, la sintassi per un percorso Amazon S3 segue una sintassi diversa appropriata per una tabella di database.

Successivamente, il DataNode componente S3 di output definisce la posizione di destinazione di output per i dati. Segue lo stesso formato del DataNode componente S3 di input, tranne il nome del componente e un percorso diverso per indicare il file di destinazione.

```
{
  "id": "S3Output",
  "type": "S3DataNode",
  "schedule": {
```

```
    "ref": "MySchedule"
  },
  "filePath": "s3://example-bucket/destination/outputfile.csv"
},
```

Risorsa

Questa è una definizione della risorsa di calcolo che esegue l'operazione di copia. In questo esempio, AWS Data Pipeline deve creare automaticamente un'istanza EC2 per eseguire l'attività di copia e terminare la risorsa dopo aver completato l'attività. I campi definiti qui controllano la creazione e la funzione dell'istanza EC2 che esegue il lavoro. EC2Resource è definito dai campi seguenti:

```
{
  "id": "MyEC2Resource",
  "type": "Ec2Resource",
  "schedule": {
    "ref": "MySchedule"
  },
  "instanceType": "m1.medium",
  "role": "DataPipelineDefaultRole",
  "resourceRole": "DataPipelineDefaultResourceRole"
},
```

Id

Il nome definito dall'utente per la pianificazione della pipeline, un'etichetta solo di riferimento.

Type (Tipo)

Il tipo di risorsa di calcolo per eseguire il lavoro, in questo caso, un'istanza EC2. Sono disponibili altri tipi di risorse, ad esempio un EmrCluster tipo.

Pianificazione

La pianificazione su cui creare questa risorsa di calcolo.

instanceType

La dimensione dell'istanza EC2 da creare. Assicurarsi che vengano impostate le dimensioni appropriate dell'istanza EC2 che meglio soddisfano il carico di lavoro che si desidera eseguire con AWS Data Pipeline. In questo caso, abbiamo impostato un'istanza m1.medium EC2. Per ulteriori informazioni sui diversi tipi di istanza e su quando utilizzarli, consulta l'argomento sui [tipi di istanza di Amazon EC2](http://aws.amazon.com/ec2/instance-types/) all'indirizzo <http://aws.amazon.com/ec2/instance-types/>.

Ruolo

Il ruolo IAM dell'account che accede alle risorse, ad esempio l'accesso a un bucket Amazon S3 per recuperare i dati.

resourceRole

Il ruolo IAM dell'account che crea le risorse, ad esempio la creazione e la configurazione di un'istanza EC2 a tuo nome. Ruolo e ResourceRole possono avere lo stesso ruolo, ma forniscono separatamente una maggiore granularità nella configurazione di sicurezza.

Attività

L'ultima sezione del file JSON è la definizione dell'attività che rappresenta il lavoro da eseguire. Questo esempio consente CopyActivity di copiare i dati da un file CSV in un bucket `http://aws.amazon.com/ec2/instance-types/` a un altro. Questo componente CopyActivity è definito dai campi seguenti:

```
{
  "id": "MyCopyActivity",
  "type": "CopyActivity",
  "runsOn": {
    "ref": "MyEC2Resource"
  },
  "input": {
    "ref": "S3Input"
  },
  "output": {
    "ref": "S3Output"
  },
  "schedule": {
    "ref": "MySchedule"
  }
}
```

Id

Il nome definito dall'utente per l'attività, un'etichetta solo di riferimento.

Type (Tipo)

Il tipo di attività da eseguire, ad esempio MyCopyActivity.

runsOn

La risorsa di calcolo che esegue il lavoro definito dall'attività. In questo esempio, forniamo un riferimento all'istanza EC2 definita in precedenza. Utilizzando il campo `runsOn`, AWS Data Pipeline crea l'istanza EC2 per l'utente. Il campo `runsOn` indica che la risorsa è disponibile nell'infrastruttura AWS, mentre il valore `workerGroup` indica che si desidera utilizzare le proprie risorse locali per eseguire il lavoro.

Input

Posizione dei dati da copiare.

Output

Dati del percorso di destinazione.

Pianificazione

La pianificazione su cui eseguire questa attività.

Caricamento e attivazione della definizione della pipeline

Devi caricare la definizione della pipeline e attivare la pipeline. Nei seguenti comandi di esempio, sostituisci *pipeline_name* con un'etichetta per la tua pipeline e *pipeline_file con il percorso completo per il file di definizione della pipeline*. `.json`

AWS CLI

Per creare la definizione della pipeline e attivare la pipeline, usa il seguente comando [create-pipeline](#). Nota l'ID della tua pipeline, perché utilizzerai questo valore con la maggior parte dei comandi CLI.

```
aws datapipeline create-pipeline --name pipeline_name --unique-id token
{
  "pipelineId": "df-00627471S0VYZEXAMPLE"
}
```

Per caricare la definizione della pipeline, usa il seguente [put-pipeline-definition](#) comando.

```
aws datapipeline put-pipeline-definition --pipeline-id df-00627471S0VYZEXAMPLE --
pipeline-definition file://MyEmrPipelineDefinition.json
```

Se la pipeline viene convalidata correttamente, il `validationErrors` campo è vuoto. È necessario esaminare eventuali avvertenze.

Per attivare la pipeline, usa il seguente comando [activate-pipeline](#).

```
aws datapipeline activate-pipeline --pipeline-id df-00627471S0VYZEXAMPLE
```

Puoi verificare che la tua pipeline appaia nell'elenco delle pipeline utilizzando il seguente comando [list-pipelines](#).

```
aws datapipeline list-pipelines
```

Esporta dati MySQL su Amazon S3 utilizzando AWS Data Pipeline

Questo tutorial illustra il processo di creazione di una pipeline di dati per copiare dati (righe) da una tabella nel database MySQL a un file CSV (valori separati da virgole) in un bucket Amazon S3 e quindi inviare una notifica Amazon SNS dopo il completamento dell'attività di copia. Utilizzerai un'istanza EC2 fornita da AWS Data Pipeline per questa attività di copia.

Oggetti della pipeline

La pipeline utilizza i seguenti oggetti:

- [CopyActivity](#)
- [Ec2Resource](#)
- [MySqlDataNode](#)
- [S3 DataNode](#)
- [SnsAlarm](#)

Indice

- [Prima di iniziare](#)
- [Copia dei dati su MySQL tramite la riga di comando](#)

Prima di iniziare

Assicurarsi di aver effettuato le operazioni seguenti.

- Completare le operazioni descritte in [Configurazione di AWS Data Pipeline](#).
- (facoltativo) Impostare un VPC per l'istanza e un gruppo di sicurezza per il VPC.

- Crea un bucket Amazon S3 come output di dati.

Per ulteriori informazioni, consulta la Guida per l'utente [di Create a bucket](#) in Amazon Simple Storage Service.

- Creare e avviare un'istanza di database MySQL come origine dati.

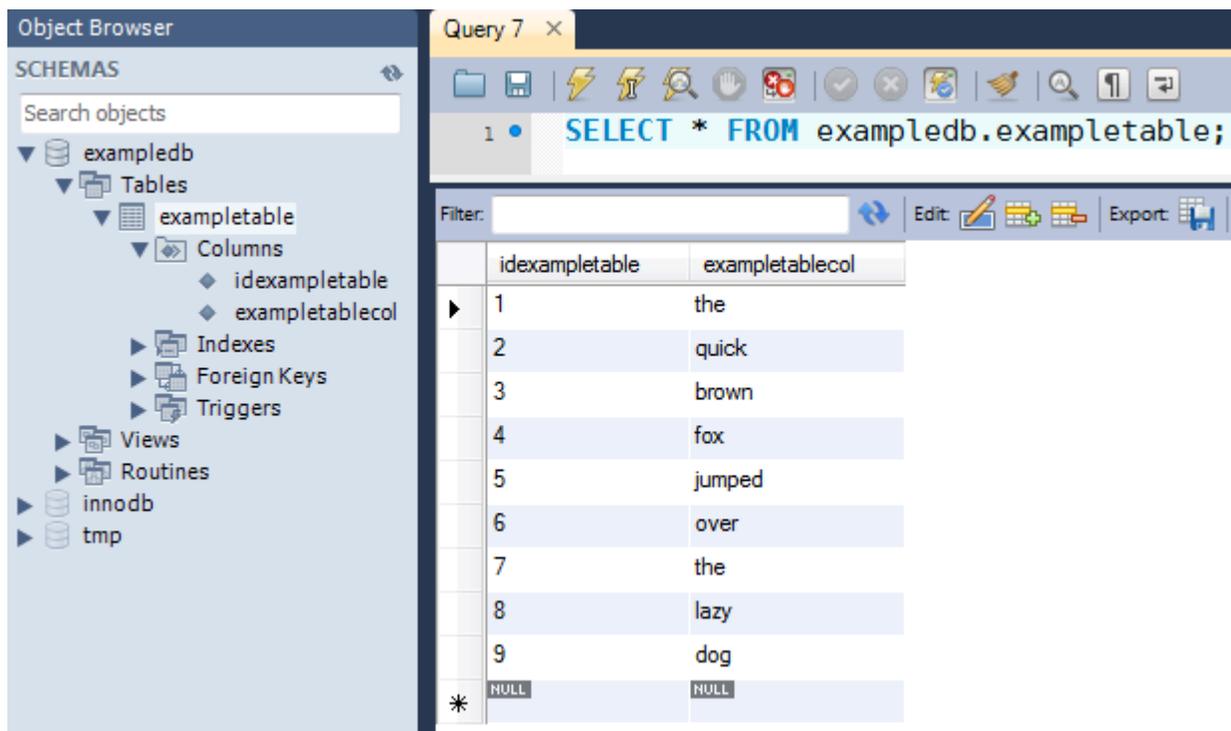
Per ulteriori informazioni, consulta [Launch a DB Instance](#) nella Guida introduttiva di Amazon RDS. Dopo aver creato un'istanza Amazon RDS, consulta [Creare una tabella](#) nella documentazione di MySQL.

Note

Annotare il nome utente e la password che hai utilizzato per creare l'istanza di MySQL. Una volta avviata l'istanza di database MySQL, annotare l'endpoint dell'istanza. Queste informazioni serviranno in seguito.

- Collegare la tua istanza di database MySQL, creare una tabella e quindi aggiungere i valori dei dati di prova alla nuova tabella creata.

A scopo illustrativo, abbiamo creato questo tutorial utilizzando una tabella di MySQL con i seguenti dati di esempio e la seguente configurazione. La seguente schermata viene da MySQL Workbench 5.2 CE:



The screenshot displays the MySQL Workbench interface. On the left, the Object Browser shows a tree view of the database schema, including 'exampledb', 'Tables', 'Columns', 'Indexes', 'Foreign Keys', 'Triggers', 'Views', and 'Routines'. The main window shows a query editor with the following SQL statement:

```
SELECT * FROM exampledb.exampletable;
```

The query results are displayed in a table with two columns: 'idexampletable' and 'exampletablecol'. The results are as follows:

	idexampletable	exampletablecol
1		the
2		quick
3		brown
4		fox
5		jumped
6		over
7		the
8		lazy
9		dog
*	NULL	NULL

Per ulteriori informazioni, consulta [Creare una tabella](#) nella documentazione di MySQL e nella [pagina dei prodotti di MySQL Workbench](#).

- Creare un argomento per l'invio di e-mail di notifica e annotare l'ARN (Amazon Resource Name) dell'argomento. Per ulteriori informazioni, consulta la Guida introduttiva alla [creazione di un argomento](#) in Amazon Simple Notification Service.
- (Facoltativo) Questo tutorial utilizza le policy del ruolo IAM di default create da AWS Data Pipeline. Se preferisci creare e configurare la policy dei ruoli IAM e le relazioni di fiducia, segui le istruzioni descritte in [Ruoli IAM per AWS Data Pipeline](#).

Copia dei dati su MySQL tramite la riga di comando

Puoi creare una pipeline per copiare dati da una tabella MySQL a un file in un bucket Amazon S3.

Prerequisiti

Prima di iniziare , devi completare le fasi seguenti:

1. Installa e configura un'interfaccia a riga di comando (CLI). Per ulteriori informazioni, consulta [Accesso a AWS Data Pipeline](#).
2. Assicurati che i ruoli IAM siano denominati DataPipelineDefaultRoleed DataPipelineDefaultResourceRoleesistano. La AWS Data Pipeline console crea questi ruoli automaticamente. Se non hai usato la AWS Data Pipeline console almeno una volta, devi creare questi ruoli manualmente. Per ulteriori informazioni, consulta [Ruoli IAM per AWS Data Pipeline](#).
3. Configura un bucket Amazon S3 e un'istanza Amazon RDS. Per ulteriori informazioni, consulta [Prima di iniziare](#).

Processi

- [Definire una pipeline in formato JSON](#)
- [Caricamento e attivazione della definizione della pipeline](#)

Definire una pipeline in formato JSON

Questo scenario di esempio mostra come utilizzare le definizioni della pipeline JSON e l'interfaccia a AWS Data Pipeline riga di comando per copiare dati (righe) da una tabella in un database MySQL a un file CSV (valori separati da virgole) in un bucket Amazon S3 a un intervallo di tempo specificato.

Questo è il file JSON completo di definizione della pipeline seguito da una spiegazione per ciascuna delle sue sezioni.

Note

È consigliabile utilizzare un editor di testo che può aiutare a verificare la sintassi di file in formato JSON e nominare il file utilizzando l'estensione del file .json.

```
{
  "objects": [
    {
      "id": "ScheduleId113",
      "startDateTime": "2013-08-26T00:00:00",
      "name": "My Copy Schedule",
      "type": "Schedule",
      "period": "1 Days"
    },
    {
      "id": "CopyActivityId112",
      "input": {
        "ref": "MySqlDataNodeId115"
      },
      "schedule": {
        "ref": "ScheduleId113"
      },
      "name": "My Copy",
      "runsOn": {
        "ref": "Ec2ResourceId116"
      },
      "onSuccess": {
        "ref": "ActionId1"
      },
      "onFail": {
        "ref": "SnsAlarmId117"
      },
      "output": {
        "ref": "S3DataNodeId114"
      },
      "type": "CopyActivity"
    },
  ]
}
```

```

    "id": "S3DataNodeId114",
    "schedule": {
      "ref": "ScheduleId113"
    },
    "filePath": "s3://example-bucket/rds-output/output.csv",
    "name": "My S3 Data",
    "type": "S3DataNode"
  },
  {
    "id": "MySQLDataNodeId115",
    "username": "my-username",
    "schedule": {
      "ref": "ScheduleId113"
    },
    "name": "My RDS Data",
    "*password": "my-password",
    "table": "table-name",
    "connectionString": "jdbc:mysql://your-sql-instance-name.id.region-name.rds.amazonaws.com:3306/database-name",
    "selectQuery": "select * from #{table}",
    "type": "SqlDataNode"
  },
  {
    "id": "Ec2ResourceId116",
    "schedule": {
      "ref": "ScheduleId113"
    },
    "name": "My EC2 Resource",
    "role": "DataPipelineDefaultRole",
    "type": "Ec2Resource",
    "resourceRole": "DataPipelineDefaultResourceRole"
  },
  {
    "message": "This is a success message.",
    "id": "ActionId1",
    "subject": "RDS to S3 copy succeeded!",
    "name": "My Success Alarm",
    "role": "DataPipelineDefaultRole",
    "topicArn": "arn:aws:sns:us-east-1:123456789012:example-topic",
    "type": "SnsAlarm"
  },
  {
    "id": "Default",
    "scheduleType": "timeseries",

```

```

    "failureAndRerunMode": "CASCADE",
    "name": "Default",
    "role": "DataPipelineDefaultRole",
    "resourceRole": "DataPipelineDefaultResourceRole"
  },
  {
    "message": "There was a problem executing #{node.name} at for period
#{node.@scheduledStartTime} to #{node.@scheduledEndTime}",
    "id": "SnsAlarmId117",
    "subject": "RDS to S3 copy failed",
    "name": "My Failure Alarm",
    "role": "DataPipelineDefaultRole",
    "topicArn": "arn:aws:sns:us-east-1:123456789012:example-topic",
    "type": "SnsAlarm"
  }
]
}

```

Nodo di dati MySQL

Il componente della `MySQLDataNode` pipeline di input definisce una posizione per i dati di input; in questo caso, un'istanza Amazon RDS. Il `MySQLDataNode` componente di input è definito dai seguenti campi:

```

{
  "id": "MySQLDataNodeId115",
  "username": "my-username",
  "schedule": {
    "ref": "ScheduleId113"
  },
  "name": "My RDS Data",
  "*password": "my-password",
  "table": "table-name",
  "connectionString": "jdbc:mysql://your-sql-instance-name.id.region-
name.rds.amazonaws.com:3306/database-name",
  "selectQuery": "select * from #{table}",
  "type": "SqlDataNode"
},

```

Id

Il nome definito dall'utente, un'etichetta solo di riferimento.

Username

Il nome utente dell'account di database che dispone di autorizzazioni sufficienti per recuperare i dati dalla tabella di database. Sostituisci *my-username* con il nome del tuo utente.

Pianificazione

Un riferimento al componente di pianificazione che abbiamo creato nelle righe precedenti del file JSON.

Nome

Il nome definito dall'utente, un'etichetta solo di riferimento.

*Password

La password per l'account di database con il prefisso asterisco per confermare che AWS Data Pipeline deve crittografare il valore della password. Sostituisci *my-password* con la password corretta per il tuo utente. Il campo password è preceduto dal carattere speciale dell'asterisco. Per ulteriori informazioni, consulta [Caratteri speciali](#).

Tabella

Il nome della tabella del database che contiene i dati da copiare. Sostituire *table_name* con il nome della tabella del database

connectionString

La stringa di connessione JDBC per la connessione CopyActivity dell'oggetto al database.

selectQuery

Una query SQL SELECT valida che specifichi quali dati copiare dalla tabella di database. Si noti che `#{table}` è un'espressione che riutilizza il nome della tabella fornito dalla variabile "tabella" nelle righe precedenti del file JSON.

Type (Tipo)

Il `SqlDataNode` tipo, che è un'istanza Amazon RDS che utilizza MySQL in questo esempio.

Note

Il tipo `MySQLDataNode` è obsoleto. Sebbene tu possa ancora utilizzarlo `MySQLDataNode`, ti consigliamo di utilizzarlo `SqlDataNode`.

Nodo dati Amazon S3

Successivamente, il componente della pipeline S3Output definisce una posizione per il file di output; in questo caso un file CSV in una posizione del bucket Amazon S3. Il DataNode componente S3 di output è definito dai seguenti campi:

```
{
  "id": "S3DataNodeId114",
  "schedule": {
    "ref": "ScheduleId113"
  },
  "filePath": "s3://example-bucket/rds-output/output.csv",
  "name": "My S3 Data",
  "type": "S3DataNode"
},
```

Id

L'ID definito dall'utente, un'etichetta solo di riferimento.

Pianificazione

Un riferimento al componente di pianificazione che abbiamo creato nelle righe precedenti del file JSON.

filePath

Il percorso ai dati associati al nodo di dati, cioè il file di output CSV in questo esempio.

Nome

Il nome definito dall'utente, un'etichetta solo di riferimento.

Type (Tipo)

Il tipo di oggetto della pipeline, che corrisponde DataNode alla posizione in cui risiedono i dati, è S3 in un bucket Amazon S3.

Risorsa

Questa è una definizione della risorsa di calcolo che esegue l'operazione di copia. In questo esempio, AWS Data Pipeline deve creare automaticamente un'istanza EC2 per eseguire l'attività di copia e terminare la risorsa dopo aver completato l'attività. I campi definiti qui controllano la creazione e la funzione dell'istanza EC2 che esegue il lavoro. EC2Resource è definito dai campi seguenti:

```
{
  "id": "Ec2ResourceId116",
  "schedule": {
    "ref": "ScheduleId113"
  },
  "name": "My EC2 Resource",
  "role": "DataPipelineDefaultRole",
  "type": "Ec2Resource",
  "resourceRole": "DataPipelineDefaultResourceRole"
},
```

Id

L'ID definito dall'utente, un'etichetta solo di riferimento.

Pianificazione

La pianificazione su cui creare questa risorsa di calcolo.

Nome

Il nome definito dall'utente, un'etichetta solo di riferimento.

Ruolo

Il ruolo IAM dell'account che accede alle risorse, ad esempio l'accesso a un bucket Amazon S3 per recuperare i dati.

Type (Tipo)

Il tipo di risorsa di calcolo per eseguire il lavoro, in questo caso, un'istanza EC2. Sono disponibili altri tipi di risorse, ad esempio un EmrCluster tipo.

resourceRole

Il ruolo IAM dell'account che crea le risorse, ad esempio la creazione e la configurazione di un'istanza EC2 a tuo nome. Ruolo e ResourceRole possono avere lo stesso ruolo, ma forniscono separatamente una maggiore granularità nella configurazione di sicurezza.

Attività

L'ultima sezione del file JSON è la definizione dell'attività che rappresenta il lavoro da eseguire. In questo caso utilizziamo un CopyActivity componente per copiare i dati da un file in un bucket Amazon S3 a un altro file. Questo componente CopyActivity è definito dai campi seguenti:

```
{
  "id": "CopyActivityId112",
  "input": {
    "ref": "MySQLDataNodeId115"
  },
  "schedule": {
    "ref": "ScheduleId113"
  },
  "name": "My Copy",
  "runsOn": {
    "ref": "Ec2ResourceId116"
  },
  "onSuccess": {
    "ref": "ActionId1"
  },
  "onFail": {
    "ref": "SnsAlarmId117"
  },
  "output": {
    "ref": "S3DataNodeId114"
  },
  "type": "CopyActivity"
},
```

Id

L'ID definito dall'utente, un'etichetta solo di riferimento

Input

Posizione dei dati MySQL da copiare

Pianificazione

La pianificazione su cui eseguire questa attività

Nome

Il nome definito dall'utente, un'etichetta solo di riferimento

runsOn

La risorsa di calcolo che esegue il lavoro definito dall'attività. In questo esempio, forniamo un riferimento all'istanza EC2 definita in precedenza. Utilizzando il campo `runsOn`, AWS Data Pipeline crea l'istanza EC2 per l'utente. Il campo `runsOn` indica che la risorsa è disponibile

nell'infrastruttura AWS, mentre il valore `workerGroup` indica che si desidera utilizzare le proprie risorse locali per eseguire il lavoro.

`onSuccess`

[SnsAlarm](#) da inviare se l'attività viene completata correttamente

`onFail`

[SnsAlarm](#) da inviare se l'attività non viene completata correttamente

Output

La posizione Amazon S3 del file di output CSV

Type (Tipo)

Il tipo di attività da eseguire.

Caricamento e attivazione della definizione della pipeline

Devi caricare la definizione della pipeline e attivare la pipeline. Nei seguenti comandi di esempio, sostituisci *pipeline_name* con un'etichetta per la tua pipeline e *pipeline_file con il percorso completo per il file di definizione della pipeline*. `.json`

AWS CLI

Per creare la definizione della pipeline e attivare la pipeline, usa il seguente comando [create-pipeline](#). Nota l'ID della tua pipeline, perché utilizzerai questo valore con la maggior parte dei comandi CLI.

```
aws datapipeline create-pipeline --name pipeline_name --unique-id token
{
  "pipelineId": "df-00627471S0VYZEXAMPLE"
}
```

Per caricare la definizione della pipeline, usa il seguente [put-pipeline-definition](#) comando.

```
aws datapipeline put-pipeline-definition --pipeline-id df-00627471S0VYZEXAMPLE --
pipeline-definition file://MyEmrPipelineDefinition.json
```

Se la pipeline viene convalidata correttamente, il `validationErrors` campo è vuoto. È necessario esaminare eventuali avvertenze.

Per attivare la pipeline, usa il seguente comando [activate-pipeline](#).

```
aws datapipeline activate-pipeline --pipeline-id df-00627471S0VYZEXAMPLE
```

Puoi verificare che la tua pipeline appaia nell'elenco delle pipeline utilizzando il seguente comando [list-pipelines](#).

```
aws datapipeline list-pipelines
```

Copia i dati su Amazon Redshift utilizzando AWS Data Pipeline

Questo tutorial illustra il processo di creazione di una pipeline che sposta periodicamente i dati da Amazon S3 ad Amazon Redshift utilizzando il modello Copy to Redshift nella AWS Data Pipeline console o un file di definizione della pipeline con l'interfaccia a riga di comando. AWS Data Pipeline

Amazon S3 è un servizio Web che consente di archiviare dati nel cloud. Per ulteriori informazioni, consultare la [Guida per l'utente di Amazon Simple archiviazione Service](#).

Amazon Redshift è un servizio di data warehouse nel cloud. Per ulteriori informazioni, consulta la [Amazon Redshift Management Guide](#).

Questo tutorial ha diversi prerequisiti. Dopo aver completato i passaggi seguenti, è possibile proseguire utilizzando la console o l'interfaccia a riga di comando.

Indice

- [Prima di iniziare: configura le opzioni COPY e di caricamento dati](#)
- [Configura Pipeline, crea un gruppo di sicurezza e crea un cluster Amazon Redshift](#)
- [Copia i dati su Amazon Redshift utilizzando la riga di comando](#)

Prima di iniziare: configura le opzioni COPY e di caricamento dati

Prima di copiare i dati su Amazon Redshift all'interno AWS Data Pipeline, assicurati di:

- Carica dati da Amazon S3.
- Configura l'COPYattività in Amazon Redshift.

Quando queste opzioni sono attive e completano correttamente un caricamento dati, trasferiscile ad AWS Data Pipeline per eseguire la copia al suo interno.

Per COPY le opzioni, consulta [COPY](#) nella Amazon Redshift Database Developer Guide.

Per i passaggi per caricare dati da Amazon S3, consulta [Caricamento dei dati da Amazon S3 nella Amazon Redshift Database Developer Guide](#).

Ad esempio, il seguente comando SQL in Amazon Redshift crea una nuova tabella denominata LISTING e copia i dati di esempio da un bucket disponibile pubblicamente in Amazon S3.

Sostituire <iam-role-arn> e la regione con la propria.

Per ulteriori informazioni su questo esempio, consulta [Load Sample Data from Amazon S3](#) nella Guida introduttiva di Amazon Redshift.

```
create table listing(  
  listid integer not null distkey,  
  sellerid integer not null,  
  eventid integer not null,  
  dateid smallint not null sortkey,  
  numtickets smallint not null,  
  priceperticket decimal(8,2),  
  totalprice decimal(8,2),  
  listtime timestamp);  
  
copy listing from 's3://awssampleduswest2/ticket/listings_pipe.txt'  
credentials 'aws_iam_role=<iam-role-arn>'  
delimiter '|' region 'us-west-2';
```

Configura Pipeline, crea un gruppo di sicurezza e crea un cluster Amazon Redshift

Per configurare il tutorial

1. Completare le operazioni descritte in [Configurazione di AWS Data Pipeline](#).
2. Creare un gruppo di sicurezza.
 - a. Aprire la console Amazon EC2.
 - b. Nel riquadro di navigazione fare clic su Security Groups (Gruppi di sicurezza).
 - c. Fare clic su Create Security Group (Crea un gruppo di sicurezza).
 - d. Specificare un nome e una descrizione per il gruppo di sicurezza.
 - e. [EC2-Classical] Seleziona No VPC per VPC.

- f. [EC2-VPC] Seleziona l'ID del VPC per VPC.
 - g. Fai clic su Create (Crea).
3. [EC2-Classic] Crea un gruppo di sicurezza del cluster Amazon Redshift e specifica il gruppo di sicurezza Amazon EC2.
 - a. Apri la console Amazon Redshift.
 - b. Nel riquadro di navigazione fare clic su Security Groups (Gruppi di sicurezza).
 - c. Fai clic su Create Cluster Security Group (Crea gruppo di sicurezza cluster).
 - d. Nella finestra di dialogo Create Cluster Security Group (Crea gruppo di sicurezza cluster) specificare un nome e una descrizione per il gruppo di sicurezza de cluster.
 - e. Fai clic sul nome del nuovo gruppo di sicurezza del cluster.
 - f. Fai clic su Add Connection Type (Aggiungi tipo di connessione).
 - g. Nella finestra di dialogo Add Connection Type (Aggiungi tipo di connessione), selezionare EC2 Security Group (Gruppo di sicurezza EC2) da Connection Type (Tipo di connessione), selezionare il gruppo di sicurezza creato da EC2 Security Group Name (Nome gruppo di sicurezza EC2), quindi fare clic su Authorize (Autorizza).
4. [EC2-VPC] Crea un gruppo di sicurezza del cluster Amazon Redshift e specifica il gruppo di sicurezza VPC.
 - a. Aprire la console Amazon EC2.
 - b. Nel riquadro di navigazione fare clic su Security Groups (Gruppi di sicurezza).
 - c. Fare clic su Create Security Group (Crea un gruppo di sicurezza).
 - d. Nella finestra di dialogo Create Security Group (Crea gruppo di sicurezza), specificare un nome e una descrizione per il gruppo di sicurezza e selezionare l'ID del tuo VPC per VPC.
 - e. Fai clic su Add Rule (Aggiungi regola). Specificare il tipo, il protocollo e l'intervallo di porte e iniziare a digitare l'ID del gruppo di sicurezza in Source (Origine). Selezionare il gruppo di sicurezza creato nel secondo passaggio.
 - f. Fai clic su Create (Crea).
5. Di seguito è riportato un riepilogo delle fasi.

Se disponi di un cluster Amazon Redshift esistente, annota l'ID del cluster.

Per creare un nuovo cluster e caricare dati di esempio, segui i passaggi in [Guida introduttiva ad Amazon Redshift](#). Per ulteriori informazioni sulla creazione di cluster, consulta [Creazione di un cluster](#) nella Amazon Redshift Management Guide.

- a. Apri la console Amazon Redshift.
- b. Fai clic su Launch Cluster (Avvia cluster).
- c. Fornire le informazioni richieste per il cluster, quindi fare clic su Continue (Continua).
- d. Fornire la configurazione del nodo, quindi fare clic su Continue (Continua).
- e. Nella pagina per ulteriori informazioni sulla configurazione, selezionare il gruppo di sicurezza del cluster creato, quindi fare clic su Continue (Continua).
- f. Esaminare le specifiche per il cluster, quindi fare clic su Launch Cluster (Avvia cluster).

Copia i dati su Amazon Redshift utilizzando la riga di comando

Questo tutorial dimostra come copiare dati da Amazon S3 ad Amazon Redshift. Creerai una nuova tabella in Amazon Redshift e la utilizzerai AWS Data Pipeline per trasferire i dati a questa tabella da un bucket Amazon S3 pubblico, che contiene dati di input di esempio in formato CSV. I log vengono salvati in un bucket Amazon S3 di tua proprietà.

Amazon S3 è un servizio Web che consente di archiviare dati nel cloud. Per ulteriori informazioni, consultare la [Guida per l'utente di Amazon Simple archiviazione Service](#). Amazon Redshift è un servizio di data warehouse nel cloud. Per ulteriori informazioni, consulta la [Amazon Redshift Management Guide](#).

Prerequisiti

Prima di iniziare , devi completare le fasi seguenti:

1. Installa e configura un'interfaccia a riga di comando (CLI). Per ulteriori informazioni, consulta [Accesso a AWS Data Pipeline](#).
2. Assicurati che i ruoli IAM siano denominati DataPipelineDefaultRole e DataPipelineDefaultResourceRole esistano. La AWS Data Pipeline console crea questi ruoli automaticamente. Se non hai usato la AWS Data Pipeline console almeno una volta, devi creare questi ruoli manualmente. Per ulteriori informazioni, consulta [Ruoli IAM per AWS Data Pipeline](#).
3. Imposta il COPY comando in Amazon Redshift, poiché è necessario che queste stesse opzioni funzionino quando esegui la copia all'interno. AWS Data Pipeline Per informazioni, consulta [Prima di iniziare: configura le opzioni COPY e di caricamento dati](#).
4. Configura un database Amazon Redshift. Per ulteriori informazioni, consulta [Configura Pipeline, crea un gruppo di sicurezza e crea un cluster Amazon Redshift](#).

Processi

- [Definire una pipeline in formato JSON](#)
- [Caricamento e attivazione della definizione della pipeline](#)

Definire una pipeline in formato JSON

Questo scenario di esempio mostra come copiare dati da un bucket Amazon S3 ad Amazon Redshift.

Questo è il file JSON completo di definizione della pipeline seguito da una spiegazione per ciascuna delle sue sezioni. È consigliabile utilizzare un editor di testo che può aiutare a verificare la sintassi di file in formato JSON e nominare il file utilizzando l'estensione del file `.json`.

```
{
  "objects": [
    {
      "id": "CSVId1",
      "name": "DefaultCSV1",
      "type": "CSV"
    },
    {
      "id": "RedshiftDatabaseId1",
      "databaseName": "dbname",
      "username": "user",
      "name": "DefaultRedshiftDatabase1",
      "*password": "password",
      "type": "RedshiftDatabase",
      "clusterId": "redshiftclusterId"
    },
    {
      "id": "Default",
      "scheduleType": "timeseries",
      "failureAndRerunMode": "CASCADE",
      "name": "Default",
      "role": "DataPipelineDefaultRole",
      "resourceRole": "DataPipelineDefaultResourceRole"
    },
    {
      "id": "RedshiftDataNodeId1",
      "schedule": {
        "ref": "ScheduleId1"
      },
      "tableName": "orders",

```

```

    "name": "DefaultRedshiftDataNode1",
    "createTableSql": "create table StructuredLogs (requestBeginTime CHAR(30)
PRIMARY KEY DISTKEY SORTKEY, requestEndTime CHAR(30), hostname CHAR(100), requestDate
varchar(20));",
    "type": "RedshiftDataNode",
    "database": {
      "ref": "RedshiftDatabaseId1"
    }
  },
  {
    "id": "Ec2ResourceId1",
    "schedule": {
      "ref": "ScheduleId1"
    },
    "securityGroups": "MySecurityGroup",
    "name": "DefaultEc2Resource1",
    "role": "DataPipelineDefaultRole",
    "logUri": "s3://myLogs",
    "resourceRole": "DataPipelineDefaultResourceRole",
    "type": "Ec2Resource"
  },
  {
    "id": "ScheduleId1",
    "startDateTime": "yyyy-mm-ddT00:00:00",
    "name": "DefaultSchedule1",
    "type": "Schedule",
    "period": "period",
    "endDateTime": "yyyy-mm-ddT00:00:00"
  },
  {
    "id": "S3DataNodeId1",
    "schedule": {
      "ref": "ScheduleId1"
    },
    "filePath": "s3://datapipeline-us-east-1/samples/hive-ads-samples.csv",
    "name": "DefaultS3DataNode1",
    "dataFormat": {
      "ref": "CSVId1"
    },
    "type": "S3DataNode"
  },
  {
    "id": "RedshiftCopyActivityId1",
    "input": {

```

```
        "ref": "S3DataNodeId1"
    },
    "schedule": {
        "ref": "ScheduleId1"
    },
    "insertMode": "KEEP_EXISTING",
    "name": "DefaultRedshiftCopyActivity1",
    "runsOn": {
        "ref": "Ec2ResourceId1"
    },
    "type": "RedshiftCopyActivity",
    "output": {
        "ref": "RedshiftDataNodeId1"
    }
}
]
```

Per ulteriori informazioni su questi oggetti, consulta la seguente documentazione.

Oggetti

- [Nodi di dati](#)
- [Risorsa](#)
- [Attività](#)

Nodi di dati

Questo esempio utilizza un nodo di dati di input, un nodo di dati di output e un database.

Nodo di dati di input

Il componente della S3DataNode pipeline di input definisce la posizione dei dati di input in Amazon S3 e il formato dei dati di input. Per ulteriori informazioni, consulta [S3 DataNode](#).

Questo componente di input è definito dai campi seguenti:

```
{
  "id": "S3DataNodeId1",
  "schedule": {
    "ref": "ScheduleId1"
  },
```

```
"filePath": "s3://datapipeline-us-east-1/samples/hive-ads-samples.csv",
"name": "DefaultS3DataNode1",
"dataFormat": {
  "ref": "CSVId1"
},
"type": "S3DataNode"
},
```

id

L'ID definito dall'utente, un'etichetta solo di riferimento.

schedule

Un riferimento al componente di pianificazione.

filePath

Il percorso ai dati associati al nodo di dati, cioè il file di input CSV in questo esempio.

name

Il nome definito dall'utente, un'etichetta solo di riferimento.

dataFormat

Un riferimento al formato dei dati dell'attività da elaborare.

Nodo dei dati di output

Il componente della `RedshiftDataNode` pipeline di output definisce una posizione per i dati di output; in questo caso, una tabella in un database Amazon Redshift. Per ulteriori informazioni, consulta [RedshiftDataNode](#). Questo componente di output è definito dai campi seguenti:

```
{
  "id": "RedshiftDataNodeId1",
  "schedule": {
    "ref": "ScheduleId1"
  },
  "tableName": "orders",
  "name": "DefaultRedshiftDataNode1",
  "createTableSql": "create table StructuredLogs (requestBeginTime CHAR(30) PRIMARY KEY DISTKEY SORTKEY, requestEndTime CHAR(30), hostname CHAR(100), requestDate varchar(20));",
```

```
"type": "RedshiftDataNode",  
"database": {  
  "ref": "RedshiftDatabaseId1"  
}  
},
```

id

L'ID definito dall'utente, un'etichetta solo di riferimento.

schedule

Un riferimento al componente di pianificazione.

tableName

Nome della tabella Amazon Redshift.

name

Il nome definito dall'utente, un'etichetta solo di riferimento.

createTableSql

Un'espressione SQL per creare la tabella nel database.

database

Un riferimento al database Amazon Redshift.

Database

Questo componente `RedshiftDatabase` è definito dai campi seguenti. Per ulteriori informazioni, consulta [RedshiftDatabase](#).

```
{  
  "id": "RedshiftDatabaseId1",  
  "databaseName": "dbname",  
  "username": "user",  
  "name": "DefaultRedshiftDatabase1",  
  "*password": "password",  
  "type": "RedshiftDatabase",  
  "clusterId": "redshiftclusterId"  
},
```

id

L'ID definito dall'utente, un'etichetta solo di riferimento.

databaseName

Il nome del database logico.

username

Il nome utente da fornire durante la connessione al database.

name

Il nome definito dall'utente, un'etichetta solo di riferimento.

password

La password per la connessione al database.

clusterId

L'ID del cluster Redshift.

Risorsa

Questa è una definizione della risorsa di calcolo che esegue l'operazione di copia. In questo esempio, AWS Data Pipeline deve creare automaticamente un'istanza EC2 per eseguire l'attività di copia e terminare l'istanza dopo aver terminato l'attività. I campi definiti qui controllano la creazione e la funzione dell'istanza che esegue il lavoro. Per ulteriori informazioni, consulta [Ec2Resource](#).

Questo componente `Ec2Resource` è definito dai campi seguenti:

```
{
  "id": "Ec2ResourceId1",
  "schedule": {
    "ref": "ScheduleId1"
  },
  "securityGroups": "MySecurityGroup",
  "name": "DefaultEc2Resource1",
  "role": "DataPipelineDefaultRole",
  "logUri": "s3://myLogs",
  "resourceRole": "DataPipelineDefaultResourceRole",
  "type": "Ec2Resource"
},
```

id

L'ID definito dall'utente, un'etichetta solo di riferimento.

schedule

La pianificazione su cui creare questa risorsa di calcolo.

securityGroups

Il gruppo di sicurezza da utilizzare per le istanze nel pool di risorse.

name

Il nome definito dall'utente, un'etichetta solo di riferimento.

role

Il ruolo IAM dell'account che accede alle risorse, ad esempio l'accesso a un bucket Amazon S3 per recuperare i dati.

logUri

Il percorso di destinazione di Amazon S3 per il backup dei log di Task Runner da. `Ec2Resource`

resourceRole

Il ruolo IAM dell'account che crea le risorse, ad esempio la creazione e la configurazione di un'istanza EC2 a tuo nome. `Role` e `ResourceRole` possono avere lo stesso ruolo, ma forniscono separatamente una maggiore granularità nella configurazione di sicurezza.

Attività

L'ultima sezione del file JSON è la definizione dell'attività che rappresenta il lavoro da eseguire. In questo caso, utilizziamo un `RedshiftCopyActivity` componente per copiare i dati da Amazon S3 ad Amazon Redshift. Per ulteriori informazioni, consulta [RedshiftCopyActivity](#).

Questo componente `RedshiftCopyActivity` è definito dai campi seguenti:

```
{
  "id": "RedshiftCopyActivityId1",
  "input": {
    "ref": "S3DataNodeId1"
  },
  "schedule": {
```

```
    "ref": "ScheduleId1"
  },
  "insertMode": "KEEP_EXISTING",
  "name": "DefaultRedshiftCopyActivity1",
  "runsOn": {
    "ref": "Ec2ResourceId1"
  },
  "type": "RedshiftCopyActivity",
  "output": {
    "ref": "RedshiftDataNodeId1"
  }
},
```

id

L'ID definito dall'utente, un'etichetta solo di riferimento.

input

Un riferimento al file sorgente di Amazon S3.

schedule

La pianificazione su cui eseguire questa attività.

insertMode

Il tipo di inserimento (KEEP_EXISTING, OVERWRITE_EXISTING o TRUNCATE).

name

Il nome definito dall'utente, un'etichetta solo di riferimento.

runsOn

La risorsa di calcolo che esegue il lavoro definito dall'attività.

output

Un riferimento alla tabella di destinazione di Amazon Redshift.

Caricamento e attivazione della definizione della pipeline

Devi caricare la definizione della pipeline e attivare la pipeline. Nei seguenti comandi di esempio, sostituisci *pipeline_name* con un'etichetta per la tua pipeline e *pipeline_file con il percorso completo per il file di definizione della pipeline*. .json

AWS CLI

Per creare la definizione della pipeline e attivare la pipeline, usa il seguente comando [create-pipeline](#). Nota l'ID della tua pipeline, perché utilizzerai questo valore con la maggior parte dei comandi CLI.

```
aws datapipeline create-pipeline --name pipeline_name --unique-id token
{
  "pipelineId": "df-00627471S0VYZEXAMPLE"
}
```

Per caricare la definizione della pipeline, usa il seguente [put-pipeline-definition](#) comando.

```
aws datapipeline put-pipeline-definition --pipeline-id df-00627471S0VYZEXAMPLE --
pipeline-definition file://MyEmrPipelineDefinition.json
```

Se la pipeline viene convalidata correttamente, il `validationErrors` campo è vuoto. È necessario esaminare eventuali avvertenze.

Per attivare la pipeline, usa il seguente comando [activate-pipeline](#).

```
aws datapipeline activate-pipeline --pipeline-id df-00627471S0VYZEXAMPLE
```

Puoi verificare che la tua pipeline appaia nell'elenco delle pipeline utilizzando il seguente comando [list-pipelines](#).

```
aws datapipeline list-pipelines
```

Funzioni ed espressioni della pipeline

Questa sezione illustra la sintassi per l'utilizzo di espressioni e funzioni in pipeline, inclusi i tipi di dati associati.

Tipi di dati di esempio

I seguenti tipi di dati possono essere impostati come valori di campi.

Tipi

- [DateTime](#)
- [Numerico](#)
- [Riferimenti agli oggetti](#)
- [Periodo](#)
- [Stringa](#)

DateTime

AWS Data Pipeline supporta la data e l'ora espresse nel formato "YYYY-MM-DDTHH:MM:SS" solo in UTC/GMT. L'esempio seguente imposta il campo `startDateTime` di un oggetto `Schedule` su 1/15/2012, 11:59 p.m., nel fuso orario UTC/GMT.

```
"startDateTime" : "2012-01-15T23:59:00"
```

Numerico

AWS Data Pipeline supporta valori sia interi sia a virgola mobile.

Riferimenti agli oggetti

Un oggetto nella definizione di pipeline. Questo può essere l'oggetto corrente, il nome di un oggetto definito altrove nella pipeline o un oggetto che elenca l'oggetto corrente in un campo, a cui si fa riferimento con la parola chiave `node`. Per ulteriori informazioni su `node`, consultare [Riferimento a campi e oggetti](#). Per ulteriori informazioni sui tipi di oggetti della pipeline, consulta [Riferimento all'oggetto pipeline](#).

Periodo

Indica con quale frequenza deve essere eseguito un evento programmato. È espresso nel formato "N [years|months|weeks|days|hours|minutes]", dove N è un valore intero positivo.

La durata minima è pari a 15 minuti, mentre la durata massima è di 3 anni.

L'esempio seguente imposta il campo `period` di un oggetto `Schedule` su 3 ore. In questo modo si crea una pianificazione che viene eseguita ogni tre ore.

```
"period" : "3 hours"
```

Stringa

Valori della stringa standard. Le stringhe devono essere racchiusi tra doppie virgolette (""). È possibile utilizzare la barra rovesciata (\) per ignorare i caratteri in una stringa. Le stringhe a più righe non sono supportate.

I seguenti esempi mostrano esempi di valori di stringhe validi per il campo `id`.

```
"id" : "My Data Object"
```

```
"id" : "My \"Data\" Object"
```

Le stringhe possono anche contenere espressioni che valutano i valori della stringa. Questi vengono inseriti nella stringa e sono delimitati con:"#{" e "}". L'esempio seguente utilizza un'espressione per inserire il nome dell'oggetto corrente in un percorso.

```
"filePath" : "s3://myBucket/#{name}.csv"
```

Per ulteriori informazioni sull'utilizzo delle espressioni, vedi [Riferimento a campi e oggetti](#) e [Valutazione delle espressioni](#).

Espressioni

Le espressioni consentono di condividere un valore negli oggetti correlati. Le espressioni vengono elaborate dal servizio Web AWS Data Pipeline in fase di runtime, accertandosi che tutte le espressioni vengano sostituite con il valore dell'espressione.

Le espressioni sono delimitate da:"#{" e "}". È possibile utilizzare un'espressione in qualsiasi oggetto di definizione della pipeline in cui una stringa è legale. Se uno slot è un riferimento o uno di tipo ID, NAME, TYPE e SPHERE, il valore non viene valutato e viene utilizzato integralmente.

La seguente espressione chiama una delle funzioni AWS Data Pipeline. Per ulteriori informazioni, consultare [Valutazione delle espressioni](#).

```
#{format(myDateTime, 'YYYY-MM-dd hh:mm:ss')}
```

Riferimento a campi e oggetti

Le espressioni possono utilizzare i campi dell'oggetto corrente in cui l'espressione esistente o i campi di un altro oggetto collegato da un riferimento.

Un formato di slot è composto da un tempo di creazione seguito dal momento della creazione dell'oggetto, ad esempio @S3BackupLocation_2018-01-31T11:05:33.

È anche possibile consultare l'esatto ID slot specificato nella definizione della pipeline, ad esempio l'ID slot della posizione di backup Amazon S3. Per fare riferimento all'ID dello slot, utilizzare `#{parent.@id}`.

In questo esempio il campo `filePath` si riferisce al campo `id` nello stesso oggetto per creare un nome di file. Il valore di `filePath` restituisce `s3://mybucket/ExampleDataNode.csv`.

```
{
  "id" : "ExampleDataNode",
  "type" : "S3DataNode",
  "schedule" : {"ref" : "ExampleSchedule"},
  "filePath" : "s3://mybucket/#{parent.@id}.csv",
  "precondition" : {"ref" : "ExampleCondition"},
  "onFail" : {"ref" : "FailureNotify"}
}
```

Per usare un campo esistente su un altro oggetto collegato da un riferimento, utilizzare la parola chiave `node`. Questa parola chiave è disponibile solo con allarmi e oggetti di preconditione.

Per continuare con l'esempio precedente, un'espressione in un `SnsAlarm` può fare riferimento alla data e all'intervallo di tempo in un `Schedule`, poiché `S3DataNode` si riferisce a entrambi.

Nello specifico, il campo `message` di `FailureNotify` può utilizzare i campi di runtime `@scheduledStartTime` e `@scheduledEndTime` di `ExampleSchedule`, poiché il campo `onFail`

di `ExampleDataNode` fa riferimento a `FailureNotify` e il relativo campo `schedule` fa riferimento a `ExampleSchedule`.

```
{
  "id" : "FailureNotify",
  "type" : "SnsAlarm",
  "subject" : "Failed to run pipeline component",
  "message": "Error for interval
#{node.@scheduledStartTime}..#{node.@scheduledEndTime}.",
  "topicArn": "arn:aws:sns:us-east-1:28619EXAMPLE:ExampleTopic"
},
```

Note

È possibile creare pipeline con dipendenze, ad esempio le attività nella pipeline che dipendono dal lavoro di altri sistemi o attività. Se la pipeline richiede alcune risorse, aggiungere tali dipendenze alla pipeline utilizzando le precondizioni associate ai nodi di dati e alle attività. In questo modo il debug della pipeline è più semplice e la pipeline stessa è più resiliente. Inoltre, mantenere le dipendenze all'interno di una singola pipeline quando è possibile, perché la risoluzione dei problemi in più pipeline è difficile.

Espressioni nidificate

AWS Data Pipeline ti consente di nidificare i valori per creare espressioni più complesse. Ad esempio, per eseguire un calcolo relativo al tempo (sottrarre 30 minuti da `scheduledStartTime`) e utilizzare il risultato in una definizione di pipeline, è possibile usare la seguente espressione in un'attività:

```
#{format(minusMinutes(@scheduledStartTime,30), 'YYYY-MM-dd hh:mm:ss')}
```

e utilizzare il prefisso `node` se l'espressione è parte di un `SnsAlarm` o Precondizione:

```
#{format(minusMinutes(node.@scheduledStartTime,30), 'YYYY-MM-dd hh:mm:ss')}
```

Elenchi

Le espressioni possono essere valutate su elenchi e su funzioni degli elenchi. Ad esempio, supponiamo che un elenco viene definito come il seguente: `"myList": ["one", "two"]`. Se

l'elenco viene utilizzato nell'espressione `#{'this is ' + myList}`, valuterà `["this is one", "this is two"]`. Se si dispone di due elenchi, Data Pipeline finirà per appiattirli nella valutazione. Ad esempio, se `myList1` è definito come `[1,2]` e `myList2` è definito come `[3,4]`, allora l'espressione `[#{myList1}, #{myList2}]` restituirà `[1,2,3,4]`.

Espressione del nodo

AWS Data Pipeline utilizza l'espressione `#{node.*}` in `SnsAlarm` o `PreCondition` per un riferimento all'oggetto padre del componente della pipeline. Poiché `SnsAlarm` e `PreCondition` sono citati da un'attività o una risorsa senza alcun riferimento da parte loro, `node` fornisce il modo per consultare il referrer. Ad esempio, la seguente definizione di pipeline dimostra come una notifica di errore può utilizzare `node` per effettuare un riferimento al proprio padre, in questo caso `ShellCommandActivity`, e includere i tempi di inizio e di fine programmati nel messaggio `SnsAlarm`. Il riferimento `scheduledStartTime` su `ShellCommandActivity` non richiede il prefisso `node` perché `scheduledStartTime` fa riferimento a se stesso.

Note

I campi preceduti dal segno AT (@) sono campi di runtime.

```
{
  "id" : "ShellOut",
  "type" : "ShellCommandActivity",
  "input" : {"ref" : "HourlyData"},
  "command" : "/home/username/xxx.sh #{@scheduledStartTime} #{@scheduledEndTime}",
  "schedule" : {"ref" : "HourlyPeriod"},
  "stderr" : "/tmp/stderr:#{@scheduledStartTime}",
  "stdout" : "/tmp/stdout:#{@scheduledStartTime}",
  "onFail" : {"ref" : "FailureNotify"},
},
{
  "id" : "FailureNotify",
  "type" : "SnsAlarm",
  "subject" : "Failed to run pipeline component",
  "message": "Error for interval
#{@node.@scheduledStartTime}..#{@node.@scheduledEndTime}.",
  "topicArn": "arn:aws:sns:us-east-1:28619EXAMPLE:ExampleTopic"
},
```

AWS Data Pipeline supporta riferimenti transitivi per i campi definiti dall'utente, ma non per i campi di runtime. Un riferimento transitivo è un riferimento tra due componenti di pipeline che dipende da un altro componente di pipeline come intermediario. L'esempio seguente mostra un riferimento a un campo transitivo definito dall'utente e un riferimento a un campo di runtime non transitivo, entrambi validi. Per ulteriori informazioni, consultare [Campi definiti dall'utente](#).

```
{
  "name": "DefaultActivity1",
  "type": "CopyActivity",
  "schedule": {"ref": "Once"},
  "input": {"ref": "s3nodeOne"},
  "onSuccess": {"ref": "action"},
  "workerGroup": "test",
  "output": {"ref": "s3nodeTwo"}
},
{
  "name": "action",
  "type": "SnsAlarm",
  "message": "S3 bucket '#{node.output.directoryPath}' succeeded at
#{node.@actualEndTime}.",
  "subject": "Testing",
  "topicArn": "arn:aws:sns:us-east-1:28619EXAMPLE:ExampleTopic",
  "role": "DataPipelineDefaultRole"
}
```

Valutazione delle espressioni

AWS Data Pipeline fornisce un set di funzioni che è possibile utilizzare per calcolare il valore di un campo. L'esempio seguente utilizza la funzione `makeDate` per impostare il campo `startDateTime` di un oggetto `Schedule` su GMT/UTC `"2011-05-24T0:00:00"`.

```
"startDateTime" : "makeDate(2011,5,24)"
```

Funzioni matematiche

Le seguenti funzioni sono disponibili per funzionare con i valori numerici.

Funzione	Descrizione
+	Addizione.

Funzione	Descrizione
	<p>Esempio: $\#{1 + 2}$</p> <p>Risultato: 3</p>
-	<p>Sottrazione.</p> <p>Esempio: $\#{1 - 2}$</p> <p>Risultato: -1</p>
*	<p>Moltiplicazione.</p> <p>Esempio: $\#{1 * 2}$</p> <p>Risultato: 2</p>
/	<p>Divisione. Se si dividono due numeri interi, il risultato è troncato.</p> <p>Esempio: $\#{1 / 2}$ Risultato: 0</p> <p>Esempio: $\#{1.0 / 2}$ Risultato: .5</p>
^	<p>Esponente.</p> <p>Esempio: $\#{2 ^ 2}$</p> <p>Risultato: 4.0</p>

Funzioni stringa

Le seguenti funzioni sono disponibili per funzionare con i valori delle stringhe.

Funzione	Descrizione
+	Concatenazione. I valori non di stringa vengono prima convertiti in stringhe.

Funzione	Descrizione
	Esempio: <code>#{ "he1" + "1o" }</code> Risultato: "hello"

Funzioni di data e ora

Le seguenti funzioni sono disponibili per funzionare con i valori `DateTime`. Per gli esempi, il valore di `myDateTime` è `May 24, 2011 @ 5:10 pm GMT`.

Note

Il formato data/ora per AWS Data Pipeline è Joda Time, che sostituisce le classi Java data e ora. Per ulteriori informazioni, consulta [Joda Time - Classe DateTimeFormat](#).

Funzione	Descrizione
<code>int day(DateTime myDateTime)</code>	Ottiene il giorno del valore <code>DateTime</code> come intero. Esempio: <code>#{ day(myDateTime) }</code> Risultato: 24
<code>int dayOfYear(DateTime myDateTime)</code>	Ottiene il giorno dell'anno del valore <code>DateTime</code> come intero. Esempio: <code>#{ dayOfYear(myDateTime) }</code> Risultato: 144
<code>DateTime firstOfMonth(DateTime myDateTime)</code>	Crea un oggetto <code>DateTime</code> per l'inizio del mese nel <code>DateTime</code> specificato.

Funzione	Descrizione
	<p>Esempio: <code>#{firstOf Month(myDateTime)}</code></p> <p>Risultato: "2011-05-01T17:10:00z"</p>
<pre>String format(DateTime myDateTime, String format)</pre>	<p>Crea un oggetto String che è il risultato della conversione del DateTime specificato utilizzando la stringa di formato specificato.</p> <p>Esempio: <code>#{format(myDateTime, 'YYYY-MM-dd HH:mm:ss z')}</code></p> <p>Risultato: "2011-05-24T17:10:00 UTC"</p>
<pre>int hour(DateTime myDateTime)</pre>	<p>Ottiene l'ora del valore DateTime come intero.</p> <p>Esempio: <code>#{hour(myDateTime)}</code></p> <p>Risultato: 17</p>
<pre>DateTime makeDate(int year,int month,int day)</pre>	<p>Crea un oggetto DateTime, in UTC, con l'anno, il mese e il giorno specificati, a mezzanotte.</p> <p>Esempio: <code>#{makeDate(2011,5,24)}</code></p> <p>Risultato: "2011-05-24T0:00:00z"</p>

Funzione	Descrizione
<pre>DateTime makeDateTime(int year,int month,int day,int hour,int minute)</pre>	<p>Crea un oggetto DateTime, in UTC, con l'anno, il mese, il giorno, l'ora e il minuto specificati.</p> <p>Esempio: <code>#{makeDateTime(2011,5,24,14,21)}</code></p> <p>Risultato: "2011-05-24T14:21:00z"</p>
<pre>DateTime midnight(DateTime myDateTime)</pre>	<p>Crea un oggetto DateTime per la mezzanotte corrente, relativo al DateTime specificato. Per esempio, se MyDateTime è 2011-05-25T17:10:00z , il risultato è come segue.</p> <p>Esempio: <code>#{midnight(myDateTime)}</code></p> <p>Risultato: "2011-05-25T0:00:00z"</p>
<pre>DateTime minusDays(DateTime myDateTime,int daysToSub)</pre>	<p>Crea un oggetto DateTime che è il risultato della sottrazione del numero di giorni specificato dal DateTime indicato.</p> <p>Esempio: <code>#{minusDays(myDateTime,1)}</code></p> <p>Risultato: "2011-05-23T17:10:00z"</p>

Funzione	Descrizione
<code>DateTime minusHours(DateTime myDateTime,int hoursToSub)</code>	<p>Crea un oggetto DateTime che è il risultato della sottrazione del numero di ore specificato dal DateTime indicato.</p> <p>Esempio: <code>#{minusHours(myDateTime,1)}</code></p> <p>Risultato: "2011-05-24T16:10:00z"</p>
<code>DateTime minusMinutes(DateTime myDateTime,int minutesToSub)</code>	<p>Crea un oggetto DateTime che è il risultato della sottrazione del numero di minuti specificato dal DateTime indicato.</p> <p>Esempio: <code>#{minusMinutes(myDateTime,1)}</code></p> <p>Risultato: "2011-05-24T17:09:00z"</p>
<code>DateTime minusMonths(DateTime myDateTime,int monthsToSub)</code>	<p>Crea un oggetto DateTime che è il risultato della sottrazione del numero di mesi specificato dal DateTime indicato.</p> <p>Esempio: <code>#{minusMonths(myDateTime,1)}</code></p> <p>Risultato: "2011-04-24T17:10:00z"</p>

Funzione	Descrizione
<code>DateTime minusWeeks(DateTime myDateTime,int weeksToSub)</code>	<p>Crea un oggetto DateTime che è il risultato della sottrazione del numero di settimane specificato dal DateTime indicato.</p> <p>Esempio: <code>#{minusWeeks(myDateTime,1)}</code></p> <p>Risultato: "2011-05-17T17:10:00z"</p>
<code>DateTime minusYears(DateTime myDateTime,int yearsToSub)</code>	<p>Crea un oggetto DateTime che è il risultato della sottrazione del numero di anni specificato dal DateTime indicato.</p> <p>Esempio: <code>#{minusYears(myDateTime,1)}</code></p> <p>Risultato: "2010-05-24T17:10:00z"</p>
<code>int minute(DateTime myDateTime)</code>	<p>Ottiene il minuto del valore DateTime come intero.</p> <p>Esempio: <code>#{minute(myDateTime)}</code></p> <p>Risultato: 10</p>
<code>int month(DateTime myDateTime)</code>	<p>Ottiene il mese del valore DateTime come intero.</p> <p>Esempio: <code>#{month(myDateTime)}</code></p> <p>Risultato: 5</p>

Funzione	Descrizione
<code>DateTime plusDays(DateTime myDateTime,int daysToAdd)</code>	<p>Crea un oggetto DateTime che è il risultato dell'addizione del numero di giorni specificato al DateTime indicato.</p> <p>Esempio: <code>#{plusDays(myDateTime,1)}</code></p> <p>Risultato: "2011-05-25T17:10:00z"</p>
<code>DateTime plusHours(DateTime myDateTime,int hoursToAdd)</code>	<p>Crea un oggetto DateTime che è il risultato dell'addizione del numero di ore specificato al DateTime indicato.</p> <p>Esempio: <code>#{plusHours(myDateTime,1)}</code></p> <p>Risultato: "2011-05-24T18:10:00z"</p>
<code>DateTime plusMinutes(DateTime myDateTime,int minutesToAdd)</code>	<p>Crea un oggetto DateTime che è il risultato dell'addizione del numero di minuti specificato al DateTime indicato.</p> <p>Esempio: <code>#{plusMinutes(myDateTime,1)}</code></p> <p>Risultato: "2011-05-24 17:11:00z"</p>

Funzione	Descrizione
<code>DateTime plusMonths(DateTime myDateTime,int monthsToAdd)</code>	<p>Crea un oggetto DateTime che è il risultato dell'addizione del numero di mesi specificato al DateTime indicato.</p> <p>Esempio: <code>#{plusMonths(myDateTime,1)}</code></p> <p>Risultato: "2011-06-24T17:10:00z"</p>
<code>DateTime plusWeeks(DateTime myDateTime,int weeksToAdd)</code>	<p>Crea un oggetto DateTime che è il risultato dell'addizione del numero di settimane specificato al DateTime indicato.</p> <p>Esempio: <code>#{plusWeeks(myDateTime,1)}</code></p> <p>Risultato: "2011-05-31T17:10:00z"</p>
<code>DateTime plusYears(DateTime myDateTime,int yearsToAdd)</code>	<p>Crea un oggetto DateTime che è il risultato dell'addizione del numero di anni specificato al DateTime indicato.</p> <p>Esempio: <code>#{plusYears(myDateTime,1)}</code></p> <p>Risultato: "2012-05-24T17:10:00z"</p>

Funzione	Descrizione
<code>DateTime sunday(DateTime myDateTime)</code>	<p>Crea un oggetto DateTime per la domenica precedente, relativo al DateTime specificato. Se il DateTime specificato è una domenica, il risultato è il DateTime specificato.</p> <p>Esempio: <code>#{sunday(myDateTime)}</code></p> <p>Risultato: "2011-05-22 17:10:00 UTC"</p>
<code>int year(DateTime myDateTime)</code>	<p>Ottiene l'anno del valore DateTime come intero.</p> <p>Esempio: <code>#{year(myDateTime)}</code></p> <p>Risultato: 2011</p>
<code>DateTime yesterday(DateTime myDateTime)</code>	<p>Crea un oggetto DateTime per il giorno precedente, relativo al DateTime specificato. Il risultato è lo stesso di <code>minusDays(1)</code>.</p> <p>Esempio: <code>#{yesterday(myDateTime)}</code></p> <p>Risultato: "2011-05-23T17:10:00z"</p>

Caratteri speciali

AWS Data Pipeline impiega determinati caratteri che hanno un significato speciale nelle definizioni di pipeline, come illustrato nella seguente tabella.

Carattere speciale	Descrizione	Esempi
@	Campo di runtime. Questo carattere è un prefisso del nome del campo per un campo che è disponibile solo quando viene eseguita una pipeline.	@actualStartTime @failureReason @resourceStatus
#	Espressione. Le espressioni sono delimitate da:"#{" e "}" e i contenuti delle parentesi graffe sono valutati da AWS Data Pipeline. Per ulteriori informazioni, consultare Espressioni .	#{format(myDateTime,'YYYY-MM-dd hh:mm:ss')} s3://mybucket/#{id}.csv
*	Campo crittografato. Questo carattere è un prefisso del nome del campo per indicare che AWS Data Pipeline deve crittografare i contenuti di questo campo in transito tra la console o l'interfaccia a riga di comando e il servizio AWS Data Pipeline.	*password

Riferimento all'oggetto pipeline

È possibile utilizzare gli oggetti e i componenti della pipeline nella definizione della pipeline.

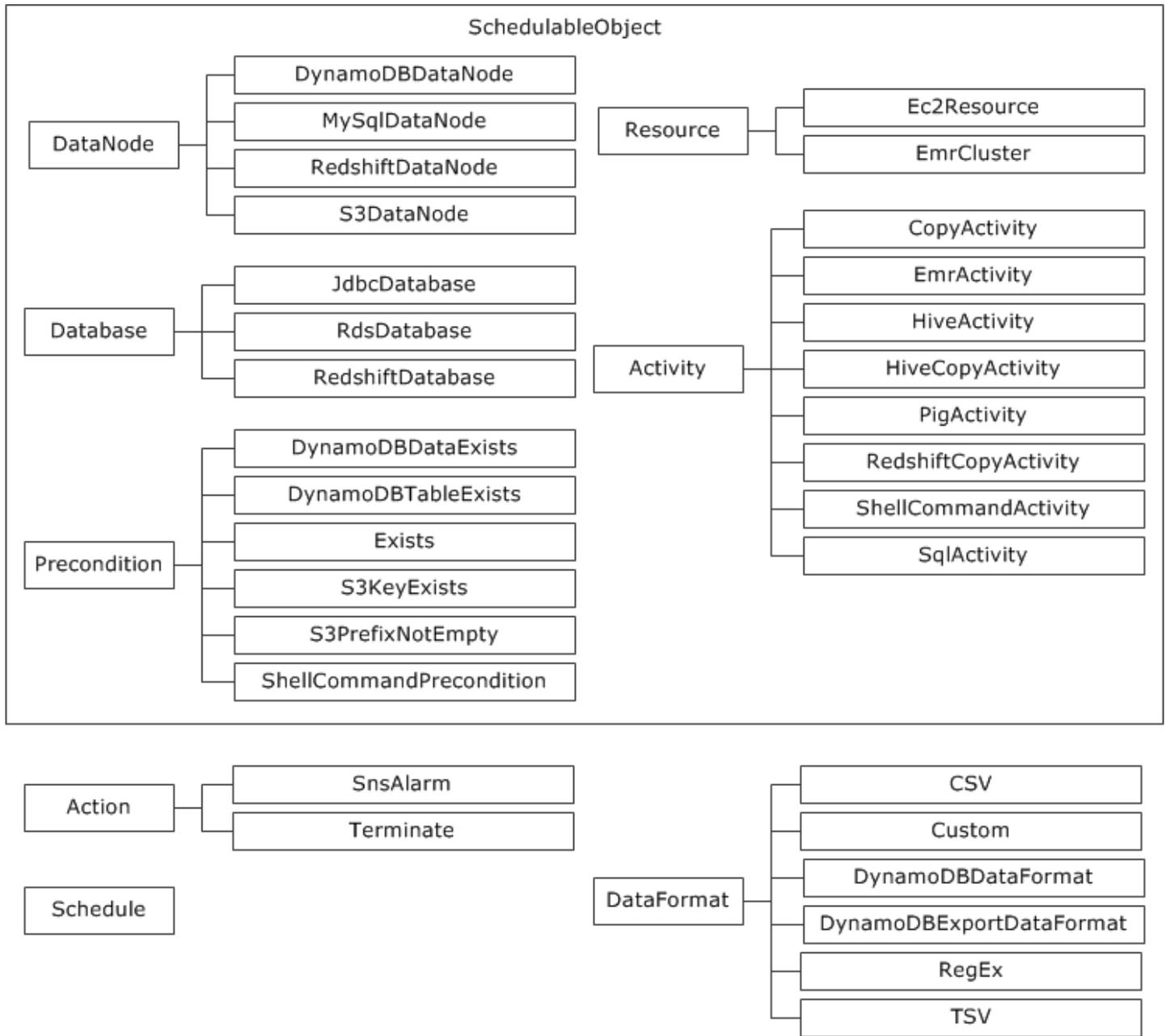
Indice

- [Nodi di dati](#)
- [Attività](#)
- [Risorse](#)
- [Precondizioni](#)
- [Database](#)
- [Formati dei dati](#)
- [Azioni](#)
- [Pianificazione](#)
- [Utilità](#)

Note

Per un'applicazione di esempio che utilizza AWS Data Pipeline Java SDK, consulta [Data Pipeline DynamoDB Export Java Sample on GitHub](#)

Di seguito è riportata la gerarchia degli oggetti per AWS Data Pipeline.



Nodi di dati

Di seguito sono elencati gli oggetti dei nodi di dati AWS Data Pipeline:

Oggetti

- [DynamoDB DataNode](#)
- [MySqlDataNode](#)
- [RedshiftDataNode](#)

- [S3 DataNode](#)
- [SqlDataNode](#)

DynamoDB DataNode

Definisce un nodo di dati utilizzando DynamoDB, che viene specificato come input per HiveActivity un oggetto or. EMRActivity

Note

L'oggetto DynamoDBDataNode non supporta la precondizione Exists.

Esempio

Di seguito è illustrato un esempio di questo tipo di oggetto. Questo oggetto fa riferimento a due altri oggetti definiti nello stesso file di definizione della pipeline. CopyPeriod è un oggetto Schedule e Ready è un oggetto di precondizione.

```
{
  "id" : "MyDynamoDBTable",
  "type" : "DynamoDBDataNode",
  "schedule" : { "ref" : "CopyPeriod" },
  "tableName" : "adEvents",
  "precondition" : { "ref" : "Ready" }
}
```

Sintassi

Campi obbligatori	Description	Tipo di slot
tableName	La tabella DynamoDB.	Stringa

Campi Object Invocation	Description	Tipo di slot
schedule	<p>Questo oggetto viene richiamato entro l'esecuzione di un intervallo di pianificazione. Gli utenti devono specificare un riferimento alla pianificazione di un altro oggetto per impostare l'ordine di esecuzione delle dipendenze per questo oggetto. Gli utenti possono soddisfare questo requisito impostando esplicitamente una pianificazione sull'oggetto, ad esempio specificando «schedule»: {"ref": "«}. DefaultSchedule Nella maggior parte dei casi, è meglio inserire il riferimento alla pianificazione nell'oggetto pipeline di default, in modo che tutti gli oggetti possano ereditare tale pianificazione. O, se la pipeline consiste di una struttura di pianificazioni (nidificate all'interno della pianificazione principale), gli utenti possono creare un oggetto padre che dispone di un riferimento alla pianificazione. Per ulteriori informazioni sulle configurazioni di pianificazione opzionali di esempio, consulta Pianificazione.</p>	<p>Oggetto di riferimento, ad esempio, «schedule»: {"ref": «} myScheduleId «}</p>

Campi opzionali	Description	Tipo di slot
attemptStatus	Lo stato segnalato più di recente dall'attività remota.	Stringa
attemptTimeout	Timeout per il completamento del lavoro in remoto. Se questo campo è impostato, un'attività remota che non viene completata entro il tempo impostato di avvio viene tentata di nuovo.	Periodo

Campi opzionali	Description	Tipo di slot
dataFormat	DataFormat per i dati descritti da questo nodo dati. Attualmente supportato per HiveActivity e HiveCopyActivity.	Oggetto di riferimento, «dataFormat»: {"ref»:» DataFormatId :»myDynamoDB «}
dependsOn	Specifica una dipendenza su un altro oggetto eseguibile	Oggetto di riferimento, ad esempio «dependsOn»: {"ref»:» «} myActivityId
failureAndRerunModo	Descrive il comportamento del nodo consumer quando le dipendenze presentano un errore o vengono di nuovo eseguite	Enumerazione
lateAfterTimeout	Il tempo trascorso dopo l'inizio della pipeline entro il quale l'oggetto deve essere completato. Viene attivato solo quando il tipo di pianificazione non è impostato su. ondemand	Periodo
maxActiveInstances	Il numero massimo di istanze attive simultanee di un componente. Le riesecuzioni non contano ai fini del numero di istanze attive.	Numero intero
maximumRetries	Numero massimo di tentativi in caso di errore	Numero intero
onFail	Un'azione da eseguire quando l'oggetto corrente ha esito negativo.	Oggetto di riferimento, ad esempio «onFail»: {"ref»:» myActionId «}
onLateAction	Azioni che devono essere attivate se un oggetto non è stato ancora pianificato o non è ancora completo.	Oggetto di riferimento, ad esempio "onLateAction«: {"ref»:» myActionId «}

Campi opzionali	Description	Tipo di slot
onSuccess	Un'operazione da eseguire quando l'oggetto corrente ha esito positivo.	Oggetto di riferimento, ad esempio «onSuccess»: {"ref":» myActionId «}
parent	Padre dell'oggetto corrente da cui saranno ereditati gli slot.	Oggetto di riferimento, ad esempio «parent»: {"ref":» myBaseObject Id "}
pipelineLogUri	L'URI S3 (come 's3://BucketName/Key/ ') per caricare i log per la pipeline.	Stringa
precondizione	Definisce eventualmente una precondizione. Un nodo dati non è contrassegnato come "READY" finché tutte le precondizioni non siano state soddisfatte.	Oggetto di riferimento, ad esempio «precondition»: {"ref":» «} myPreconditionId
readThroughputPercent	Imposta la percentuale di operazioni di lettura per mantenere il tasso di throughput assegnato di DynamoDB nell'intervallo allocato per la tabella. Il valore è compreso tra 0,1 e 1,0 inclusi.	Doppio
Regione	Il codice per la regione in cui esiste la tabella DynamoDB. Ad esempio, us-east-1. Viene utilizzato HiveActivity quando esegue lo staging per le tabelle DynamoDB in Hive.	Enumerazione
reportProgressTimeout	Timeout per chiamate successive di attività in remoto a reportProgress. Se impostato, le attività in remoto che non presentano avanzamenti nel periodo specificato potrebbero essere considerate bloccate e sono quindi oggetto di un altro tentativo.	Periodo

Campi opzionali	Description	Tipo di slot
retryDelay	La durata del timeout tra due tentativi.	Periodo
runsOn	Le risorse di calcolo per eseguire l'attività o il comando. Ad esempio, un'istanza Amazon EC2 o un cluster Amazon EMR.	Oggetto di riferimento, ad esempio «runOn»: {"ref":» myResourceId «}
scheduleType	Il tipo di pianificazione consente di specificare se gli oggetti nella definizione di pipeline devono essere programmati all'inizio o alla fine dell'intervallo. Time Series Style Scheduling significa che le istanze vengono programmate al termine di ogni intervallo e Cron Style Scheduling significa che le istanze vengono programmate all'inizio di ogni intervallo. Una pianificazione on demand consente di eseguire una pipeline una sola volta, per attivazione. Questo significa che non è necessario clonare o ricreare la pipeline per eseguirla di nuovo. Se utilizzi una pianificazione on demand, devi specificarlo nell'oggetto predefinito e deve essere l'unico scheduleType specificato per gli oggetti della pipeline. Per utilizzare le pipeline su richiesta, è sufficiente chiamare l'ActivatePipeline operazione per ogni esecuzione e successiva. I valori sono: cron, ondemand e timeseries.	Enumerazione
workerGroup	Il gruppo di lavoro. Utilizzato per le attività di routing. Se si fornisce un valore runsOn e workerGroup esiste, workerGroup verrà ignorato.	Stringa

Campi opzionali	Description	Tipo di slot
writeThroughputPercent	Imposta la percentuale di operazioni di scrittura per mantenere il tasso di throughput assegnato di DynamoDB nell'intervallo allocato per la tabella. Il valore è compreso tra 0,1 e 1,0 inclusi.	Doppio

Campi Runtime	Description	Tipo di slot
@activeInstances	Elenco di oggetti di istanze attive attualmente programmate.	Oggetto di riferimento, ad esempio «activeInstances»: {"ref»:» Id "myRunnableObject
@actualEndTime	L'ora in cui è terminata l'esecuzione di questo oggetto.	DateTime
@actualStartTime	L'ora in cui è stata avviata l'esecuzione di questo oggetto.	DateTime
cancellationReason	CancellationReason se questo oggetto è stato annullato.	Stringa
@cascadeFailedOn	Descrizione della catena di dipendenza che ha generato l'errore dell'oggetto.	Oggetto di riferimento, ad esempio "cascadeFailedOn«: {"ref»:» myRunnableObject Id "}
emrStepLog	Log della fase EMR disponibili solo sui tentativi delle attività EMR	Stringa
errorId	ErrorId se l'oggetto non è riuscito.	Stringa
errorMessage	ErrorMessage se l'oggetto non è riuscito.	Stringa

Campi Runtime	Description	Tipo di slot
errorStackTrace	Traccia dello stack di errore se l'oggetto non è riuscito.	Stringa
@finishedTime	L'ora in cui è terminata l'esecuzione di questo oggetto.	DateTime
hadoopJobLog	Log delle attività Hadoop disponibili per le attività basate su EMR.	Stringa
@healthStatus	Lo stato di integrità dell'oggetto che riflette l'esito positivo o negativo dell'ultima istanza dell'oggetto che ha raggiunto lo stato di un'istanza terminata.	Stringa
@healthStatusFromInstanceid	Id dell'ultimo oggetto dell'istanza che ha raggiunto lo stato terminato.	Stringa
@healthStatusUpdated Ora	L'ora in cui lo stato di integrità è stato aggiornato o l'ultima volta.	DateTime
hostname	Il nome host del client che si è aggiudicato il tentativo dell'attività.	Stringa
@lastDeactivatedTime	L'ora in cui l'oggetto è stato disattivato.	DateTime
@latestCompletedRun Ora	L'orario dell'esecuzione più recente durante il quale l'esecuzione è stata completata.	DateTime
@latestRunTime	L'orario dell'esecuzione più recente durante il quale l'esecuzione è stata pianificata.	DateTime
@nextRunTime	L'orario dell'esecuzione da programmare come successiva.	DateTime
reportProgressTime	Il periodo di tempo più recente in cui l'attività remota ha segnalato un progresso.	DateTime

Campi Runtime	Description	Tipo di slot
@scheduledEndTime	L'orario di termine della pianificazione per un oggetto	DateTime
@scheduledStartTime	L'orario di inizio della pianificazione per l'oggetto	DateTime
@status	Lo stato di questo oggetto.	Stringa
@version	Versione della pipeline con cui l'oggetto è stato creato.	Stringa
@waitingOn	Descrizione dell'elenco di dipendenze per cui questo oggetto è in attesa.	Oggetto di riferimento, ad esempio «waitingOn»: {"ref":» myRunnableObject Id "}

Campi di sistema	Description	Tipo di slot
@error	Errore che descrive il formato oggetto errato	Stringa
@pipelineId	L'id della pipeline a cui appartiene questo oggetto	Stringa
@sphere	La sfera di un oggetto indica la propria posizione nel ciclo di vita: i Component Objects generano Instance Objects che eseguono Attempt Objects	Stringa

MySQLDataNode

Definisce un nodo dati utilizzando MySQL.

Note

Il tipo `MySQLDataNode` è obsoleto. Consigliamo di utilizzare invece [SqlDataNode](#).

Esempio

Di seguito è illustrato un esempio di questo tipo di oggetto. Questo oggetto fa riferimento a due altri oggetti definiti nello stesso file di definizione della pipeline. `CopyPeriod` è un oggetto `Schedule` e `Ready` è un oggetto di preconditione.

```
{
  "id" : "Sql Table",
  "type" : "MySQLDataNode",
  "schedule" : { "ref" : "CopyPeriod" },
  "table" : "adEvents",
  "username": "user_name",
  "*password": "my_password",
  "connectionString": "jdbc:mysql://mysqlinstance-rds.example.us-east-1.rds.amazonaws.com:3306/database_name",
  "selectQuery" : "select * from #{table} where eventTime >=
'#{@scheduledStartTime.format('YYYY-MM-dd HH:mm:ss')}' and eventTime <
'#{@scheduledEndTime.format('YYYY-MM-dd HH:mm:ss')}'",
  "precondition" : { "ref" : "Ready" }
}
```

Sintassi

Campi obbligatori	Description	Tipo di slot
table	Nome della tabella nel database MySQL.	Stringa

Campi Object Invocation	Description	Tipo di slot
schedule	Questo oggetto viene richiamato entro l'esecuzione di un intervallo di pianificazione. Gli utenti devono specificare un riferimento alla	Oggetto di riferimento, ad esempio

Campi Object Invocation	Description	Tipo di slot
	<p>pianificazione di un altro oggetto per impostare l'ordine di esecuzione delle dipendenze per questo oggetto. Gli utenti possono soddisfare questo requisito impostando esplicitamente una pianificazione sull'oggetto, ad esempio specificando «schedule»: {"ref": "«}. DefaultSchedule Nella maggior parte dei casi, è meglio inserire il riferimento alla pianificazione nell'oggetto pipeline di default, in modo che tutti gli oggetti possano ereditare tale pianificazione. O, se la pipeline consiste di una struttura di pianificazioni (nidificate all'interno della pianificazione principale), gli utenti possono creare un oggetto padre che dispone di un riferimento alla pianificazione. Per ulteriori informazioni sulle configurazioni di pianificazione opzionali di esempio, consulta https://docs.aws.amazon.com/datapipeline/latest/DeveloperGuide/dp-object-schedule.html</p>	<p>«schedule»: {"ref": «} «myScheduleId}</p>
Campi opzionali	Description	Tipo di slot
attemptStatus	Lo stato segnalato più di recente dall'attività remota.	Stringa
attemptTimeout	Timeout per il completamento del lavoro in remoto. Se questo campo è impostato, un'attività remota che non viene completata entro il tempo impostato di avvio viene tentata di nuovo.	Periodo
createTableSql	Un'espressione SQL per creare la tabella che crea la tabella.	Stringa

Campi opzionali	Description	Tipo di slot
database	Nome del database.	Oggetto di riferimento, ad esempio «database»: {"ref":» myDatabaseId «}
dependsOn	Specifica una dipendenza su un altro oggetto eseguibile.	Oggetto di riferimento, ad esempio «dependsOn»: {"ref":» myActivityId «}
failureAndRerunModo	Descrive il comportamento del nodo consumer quando le dipendenze presentano un errore o vengono di nuovo eseguite.	Enumerazione
insertQuery	Un'istruzione SQL per inserire dati nella tabella.	Stringa
lateAfterTimeout	Il tempo trascorso dopo l'inizio della pipeline entro il quale l'oggetto deve essere completato. Viene attivato solo quando il tipo di pianificazione non è impostato su. ondemand	Periodo
maxActiveInstances	Il numero massimo di istanze attive simultanee di un componente. Le riesecuzioni non contano ai fini del numero di istanze attive.	Numero intero
maximumRetries	Numero massimo di tentativi in caso di errore	Numero intero
onFail	Un'azione da eseguire quando l'oggetto corrente ha esito negativo.	Oggetto di riferimento, ad esempio «onFail»: {"ref":» myActionId «}
onLateAction	Azioni che devono essere attivate se un oggetto non è stato ancora pianificato o non è ancora completo.	Oggetto di riferimento, ad esempio "onLateAction«: {"ref":» myActionId «}

Campi opzionali	Description	Tipo di slot
onSuccess	Un'operazione da eseguire quando l'oggetto corrente ha esito positivo.	Oggetto di riferimento, ad esempio «onSuccess»: {"ref":» myActionId «}
parent	Padre dell'oggetto corrente da cui saranno ereditati gli slot.	Oggetto di riferimento, ad esempio «parent»: {"ref":» myBaseObject Id "}
pipelineLogUri	L'URI S3 (come 's3://BucketName/Key/ ') per caricare i log per la pipeline.	Stringa
precondizione	Definisce eventualmente una precondizione. Un nodo dati non è contrassegnato come "READY" finché tutte le precondizioni non siano state soddisfatte.	Oggetto di riferimento, ad esempio «precondition»: {"ref":» «} myPreconditionId
reportProgressTimeout	Timeout per chiamate successive di attività in remoto a reportProgress. Se impostato, le attività in remoto che non presentano avanzamenti nel periodo specificato potrebbero essere considerate bloccate e sono quindi oggetto di un altro tentativo.	Periodo
retryDelay	La durata del timeout tra due tentativi.	Periodo
runsOn	Le risorse di calcolo per eseguire l'attività o il comando. Ad esempio, un'istanza Amazon EC2 o un cluster Amazon EMR.	Oggetto di riferimento, ad esempio «runson»: {"ref":» myResourceId «}

Campi opzionali	Description	Tipo di slot
scheduleType	Il tipo di pianificazione consente di specificare se gli oggetti nella definizione di pipeline devono essere programmati all'inizio o alla fine dell'intervallo. Time Series Style Scheduling significa che le istanze vengono programmate al termine di ogni intervallo e Cron Style Scheduling significa che le istanze vengono programmate all'inizio di ogni intervallo. Una pianificazione on demand consente di eseguire una pipeline una sola volta, per attivazione. Questo significa che non è necessario clonare o ricreare la pipeline per eseguirla di nuovo. Se utilizzi una pianificazione on demand, devi specificarlo nell'oggetto predefinito e deve essere l'unico scheduleType specificato per gli oggetti della pipeline. Per utilizzare le pipeline su richiesta, è sufficiente chiamare l'ActivatePipeline operazione per ogni esecuzione e successiva. I valori sono: cron, ondemand e timeseries.	Enumerazione
schemaName	Il nome dello schema in cui è presente la tabella.	Stringa
selectQuery	Un'istruzione SQL per recuperare i dati dalla tabella.	Stringa
workerGroup	Il gruppo di lavoro. Utilizzato per le attività di routing. Se si fornisce un valore runsOn e workerGroup esiste, workerGroup verrà ignorato.	Stringa

Campi Runtime	Description	Tipo di slot
@activeInstances	Elenco di oggetti di istanze attive attualmente programmate.	Oggetto di riferimento, ad esempio «activeInstances»: {"ref»:» Id "} myRunnableObject
@actualEndTime	L'ora in cui è terminata l'esecuzione di questo oggetto.	DateTime
@actualStartTime	L'ora in cui è stata avviata l'esecuzione di questo oggetto.	DateTime
cancellationReason	CancellationReason se questo oggetto è stato annullato.	Stringa
@cascadeFailedOn	Descrizione della catena di dipendenza che ha generato l'errore dell'oggetto.	Oggetto di riferimento, ad esempio "cascadeFailedOn«: {" ref»:» myRunnableObject Id "}
emrStepLog	Log della fase EMR disponibili solo sui tentativi delle attività EMR	Stringa
errorId	ErrorId se l'oggetto non è riuscito.	Stringa
errorMessage	ErrorMessage se l'oggetto non è riuscito.	Stringa
errorStackTrace	Traccia dello stack di errore se l'oggetto non è riuscito.	Stringa
@finishedTime	L'ora in cui è terminata l'esecuzione di questo oggetto.	DateTime
hadoopJobLog	Log delle attività Hadoop disponibili per le attività basate su EMR.	Stringa
@healthStatus	Lo stato di integrità dell'oggetto che riflette l'esito positivo o negativo dell'ultima istanza	Stringa

Campi Runtime	Description	Tipo di slot
	dell'oggetto che ha raggiunto lo stato di un'istanza terminata.	
@healthStatusFromInstanceid	Id dell'ultimo oggetto dell'istanza che ha raggiunto lo stato terminato.	Stringa
@healthStatusUpdated Ora	L'ora in cui lo stato di integrità è stato aggiornato o l'ultima volta.	DateTime
hostname	Il nome host del client che si è aggiudicato il tentativo dell'attività.	Stringa
@lastDeactivatedTime	L'ora in cui l'oggetto è stato disattivato.	DateTime
@latestCompletedRun Ora	L'orario dell'esecuzione più recente durante il quale l'esecuzione è stata completata.	DateTime
@latestRunTime	L'orario dell'esecuzione più recente durante il quale l'esecuzione è stata pianificata.	DateTime
@nextRunTime	L'orario dell'esecuzione da programmare come successiva.	DateTime
reportProgressTime	Il periodo di tempo più recente in cui l'attività remota ha segnalato un progresso.	DateTime
@scheduledEndTime	L'orario di termine della pianificazione per un oggetto.	DateTime
@scheduledStartTime	L'orario di inizio della pianificazione per l'oggetto.	DateTime
@status	Lo stato di questo oggetto.	Stringa
@version	Versione della pipeline con cui l'oggetto è stato creato.	Stringa

Campi Runtime	Description	Tipo di slot
@waitingOn	Descrizione dell'elenco di dipendenze per cui questo oggetto è in attesa.	Oggetto di riferimento, ad esempio «waitingOn»: {"ref":» myRunnableObject Id "}
Campi di sistema	Description	Tipo di slot
@error	Errore che descrive il formato oggetto errato.	Stringa
@pipelineId	L'id della pipeline a cui appartiene questo oggetto.	Stringa
@sphere	La sfera di un oggetto indica la propria posizione nel ciclo di vita: i Component Objects generano Instance Objects che eseguono Attempt Objects.	Stringa

Vedi anche

- [S3 DataNode](#)

RedshiftDataNode

Definisce un nodo di dati utilizzando Amazon Redshift. `RedshiftDataNode` rappresenta le proprietà dei dati all'interno di un database, ad esempio una tabella di dati, utilizzata dalla pipeline.

Esempio

Di seguito è illustrato un esempio di questo tipo di oggetto.

```
{
  "id" : "MyRedshiftDataNode",
  "type" : "RedshiftDataNode",
```

```

"database": { "ref": "MyRedshiftDatabase" },
"tableName": "adEvents",
"schedule": { "ref": "Hour" }
}

```

Sintassi

Campi obbligatori	Description	Tipo di slot
database	Il database in cui risiede la tabella.	Oggetto di riferimento, ad esempio «database»: {"ref":» myRedshiftDatabase Id "}
tableName	Nome della tabella Amazon Redshift. La tabella viene creata se non esiste già e se l'hai fornita createTableSql.	Stringa

Campi Object Invocation	Description	Tipo di slot
schedule	Questo oggetto viene richiamato entro l'esecuzione di un intervallo di pianificazione. Gli utenti devono specificare un riferimento alla pianificazione di un altro oggetto per impostare l'ordine di esecuzione delle dipendenze per questo oggetto. Gli utenti possono soddisfare questo requisito impostando esplicitamente una pianificazione sull'oggetto, ad esempio specificando «schedule»: {"ref": "DefaultSchedule"}. Nella maggior parte dei casi, è meglio inserire il riferimento alla pianificazione nell'oggetto pipeline di default, in modo che tutti gli oggetti possano ereditare tale pianificazione. O, se la pipeline consiste di una struttura di	Oggetto di riferimento, ad esempio «schedule»: {"ref":» «myScheduleId}

Campi Object Invocation	Description	Tipo di slot
	<p>pianificazioni (nidificate all'interno della pianificazione principale), gli utenti possono creare un oggetto padre che dispone di un riferimento alla pianificazione. Per ulteriori informazioni sulle configurazioni di pianificazione opzionali di esempio, consulta https://docs.aws.amazon.com/datapipeline/latest/DeveloperGuide/dp-object-schedule.html</p>	
Campi opzionali	Description	Tipo di slot
attemptStatus	Lo stato segnalato più di recente dall'attività remota.	Stringa
attemptTimeout	Timeout per il completamento del lavoro in remoto. Se questo campo è impostato, un'attività remota che non viene completata entro il tempo impostato di avvio viene tentata di nuovo.	Periodo
createTableSql	Un'espressione SQL per creare la tabella nel database. Si consiglia di specificare lo schema in cui creare la tabella, ad esempio: CREATE TABLE mySchema.myTable (bestColumn varchar (25) primary key distkey, integer sortKey). numberOfWins AWS Data Pipeline segue lo script nel createTableSql campo se la tabella, specificata da TableName, non esiste nello schema specificato dal campo SchemaName. Ad esempio, se si specifica SchemaName come mySchema ma non si include mySchema nel createTableSql campo, la tabella viene creata nello schema sbagliato	Stringa

Campi opzionali	Description	Tipo di slot
	(per impostazione predefinita, verrebbe creata in PUBLIC). Questo avviene perché AWS Data Pipeline non analizza le istruzioni CREATE TABLE.	
dependsOn	Specifica una dipendenza su un altro oggetto eseguibile	Oggetto di riferimento, ad esempio «dependsOn»: {"ref":» «} myActivityId
failureAndRerunMode	Descrive il comportamento del nodo consumer quando le dipendenze presentano un errore o vengono di nuovo eseguite	Enumerazione
lateAfterTimeout	Il tempo trascorso dopo l'inizio della pipeline entro il quale l'oggetto deve essere completato. Viene attivato solo quando il tipo di pianificazione non è impostato su. ondemand	Periodo
maxActiveInstances	Il numero massimo di istanze attive simultanee di un componente. Le riesecuzioni non contano ai fini del numero di istanze attive.	Numero intero
maximumRetries	Numero massimo di tentativi in caso di errore.	Numero intero
onFail	Un'azione da eseguire quando l'oggetto corrente ha esito negativo.	Oggetto di riferimento, ad esempio «onFail»: {"ref":» myActionId «}
onLateAction	Azioni che devono essere attivate se un oggetto non è stato ancora pianificato o non è ancora completo.	Oggetto di riferimento, ad esempio "onLateAction«: {"ref":» myActionId «}

Campi opzionali	Description	Tipo di slot
onSuccess	Un'operazione da eseguire quando l'oggetto corrente ha esito positivo.	Oggetto di riferimento, ad esempio «onSuccess»: {"ref":» myActionId «}
parent	Padre dell'oggetto corrente da cui saranno ereditati gli slot.	Oggetto di riferimento, ad esempio «parent»: {"ref":» myBaseObject Id "}
pipelineLogUri	L'URI S3 (come 's3://BucketName/Key/ ') per caricare i log per la pipeline.	Stringa
precondizione	Definisce eventualmente una precondizione. Un nodo dati non è contrassegnato come "READY" finché tutte le precondizioni non siano state soddisfatte.	Oggetto di riferimento, ad esempio «precondition»: {"ref":» «} myPreconditionId
primaryKeys	Se non si specificano le primaryKeys per una tabella di destinazione in RedShiftCopyActivity , è possibile specificare un elenco di colonne utilizzando le primaryKeys che agiranno come mergeKey. Tuttavia, se hai una PrimaryKey esistente definita in una tabella Amazon Redshift, questa impostazione sostituisce la chiave esistente.	Stringa
reportProgressTimeout	Timeout per chiamate successive di attività in remoto a reportProgress. Se impostato , le attività in remoto che non presentano avanzamenti nel periodo specificato potrebbero essere considerate bloccate e sono quindi oggetto di un altro tentativo.	Periodo
retryDelay	La durata del timeout tra due tentativi.	Periodo

Campi opzionali	Description	Tipo di slot
runsOn	Le risorse di calcolo per eseguire l'attività o il comando. Ad esempio, un'istanza Amazon EC2 o un cluster Amazon EMR.	Oggetto di riferimento, ad esempio «runSon»: {"ref":» myResourceId «}
scheduleType	Il tipo di pianificazione consente di specificare se gli oggetti nella definizione di pipeline devono essere programmati all'inizio o alla fine dell'intervallo. Time Series Style Scheduling significa che le istanze vengono programmate al termine di ogni intervallo e Cron Style Scheduling significa che le istanze vengono programmate all'inizio di ogni intervallo. Una pianificazione on demand consente di eseguire una pipeline una sola volta, per attivazione. Questo significa che non è necessario clonare o ricreare la pipeline per eseguirla di nuovo. Se utilizzi una pianificazione on demand, devi specificarlo nell'oggetto predefinito e deve essere l'unico scheduleType specificato per gli oggetti della pipeline. Per utilizzare le pipeline su richiesta, è sufficiente chiamare l'ActivatePipeline operazione per ogni esecuzione e successiva. I valori sono: cron, ondemand e timeseries.	Enumerazione
schemaName	Questo campo facoltativo specifica il nome dello schema per la tabella di Amazon Redshift. Se non è specificato, il nome dello schema è PUBLIC, che è lo schema predefinito in Amazon Redshift. Per ulteriori informazioni, consulta la Guida per sviluppatori del database di Amazon Redshift.	Stringa

Campi opzionali	Description	Tipo di slot
workerGroup	Il gruppo di lavoro. Utilizzato per le attività di routing. Se si fornisce un valore runsOn e workerGroup esiste, workerGroup verrà ignorato.	Stringa
Campi Runtime	Description	Tipo di slot
@activeInstances	Elenco di oggetti di istanze attive attualmente programmate.	Oggetto di riferimento, ad esempio «activeInstances»: {"ref":» Id "} myRunnableObject
@actualEndTime	L'ora in cui è terminata l'esecuzione di questo oggetto.	DateTime
@actualStartTime	L'ora in cui è stata avviata l'esecuzione di questo oggetto.	DateTime
cancellationReason	CancellationReason se questo oggetto è stato annullato.	Stringa
@cascadeFailedOn	Descrizione della catena di dipendenza che ha generato l'errore dell'oggetto.	Oggetto di riferimento, ad esempio "cascadeFailedOn«: {" ref":» myRunnableObject Id "}
emrStepLog	Log della fase EMR disponibili solo sui tentativi delle attività EMR	Stringa
errorId	ErrorId se l'oggetto non è riuscito.	Stringa
errorMessage	ErrorMessage se l'oggetto non è riuscito.	Stringa

Campi Runtime	Description	Tipo di slot
errorStackTrace	Traccia dello stack di errore se l'oggetto non è riuscito.	Stringa
@finishedTime	L'ora in cui è terminata l'esecuzione di questo oggetto.	DateTime
hadoopJobLog	Log delle attività Hadoop disponibili per le attività basate su EMR.	Stringa
@healthStatus	Lo stato di integrità dell'oggetto che riflette l'esito positivo o negativo dell'ultima istanza dell'oggetto che ha raggiunto lo stato di un'istanza terminata.	Stringa
@healthStatusFromInstanceid	Id dell'ultimo oggetto dell'istanza che ha raggiunto lo stato terminato.	Stringa
@healthStatusUpdated Ora	L'ora in cui lo stato di integrità è stato aggiornato o l'ultima volta.	DateTime
hostname	Il nome host del client che si è aggiudicato il tentativo dell'attività.	Stringa
@lastDeactivatedTime	L'ora in cui l'oggetto è stato disattivato.	DateTime
@latestCompletedRun Ora	L'orario dell'esecuzione più recente durante il quale l'esecuzione è stata completata.	DateTime
@latestRunTime	L'orario dell'esecuzione più recente durante il quale l'esecuzione è stata pianificata.	DateTime
@nextRunTime	L'orario dell'esecuzione da programmare come successiva.	DateTime
reportProgressTime	Il periodo di tempo più recente in cui l'attività remota ha segnalato un progresso.	DateTime

Campi Runtime	Description	Tipo di slot
@scheduledEndTime	L'orario di termine della pianificazione per un oggetto	DateTime
@scheduledStartTime	L'orario di inizio della pianificazione per l'oggetto	DateTime
@status	Lo stato di questo oggetto.	Stringa
@version	Versione della pipeline con cui l'oggetto è stato creato.	Stringa
@waitingOn	Descrizione dell'elenco di dipendenze per cui questo oggetto è in attesa.	Oggetto di riferimento, ad esempio «waitingOn»: {"ref":» myRunnableObject Id "}

Campi di sistema	Description	Tipo di slot
@error	Errore che descrive il formato oggetto errato.	Stringa
@pipelineId	L'id della pipeline a cui appartiene questo oggetto.	Stringa
@sphere	La sfera di un oggetto indica la propria posizione nel ciclo di vita: i Component Objects generano Instance Objects che eseguono Attempt Objects.	Stringa

S3 DataNode

Definisce un nodo di dati utilizzando Amazon S3. Per impostazione predefinita, S3 DataNode utilizza la crittografia lato server. Se desideri disabilitarlo, imposta EncryptionType s3 su NONE.

Note

Quando si utilizza un `S3DataNode` come input `CopyActivity`, solo i formati di dati CSV e TSV sono supportati.

Esempio

Di seguito è illustrato un esempio di questo tipo di oggetto. Questo oggetto fa riferimento a un altro oggetto definito nello stesso file di definizione della pipeline. `CopyPeriod` è un oggetto `Schedule`.

```
{
  "id" : "OutputData",
  "type" : "S3DataNode",
  "schedule" : { "ref" : "CopyPeriod" },
  "filePath" : "s3://myBucket/#{@scheduledStartTime}.csv"
}
```

Sintassi

Campi Object Invocation	Description	Tipo di slot
schedule	Questo oggetto viene richiamato entro l'esecuzione di un intervallo di pianificazione. Gli utenti devono specificare un riferimento alla pianificazione di un altro oggetto per impostare l'ordine di esecuzione delle dipendenze per questo oggetto. Gli utenti possono soddisfare questo requisito impostando esplicitamente una pianificazione sull'oggetto, ad esempio specificando «schedule»: {"ref": "DefaultSchedule"}. Nella maggior parte dei casi, è meglio inserire il riferimento alla pianificazione nell'oggetto pipeline di default, in modo che tutti gli oggetti possano ereditare tale pianificazione. O, se la pipeline consiste di una struttura di pianificazioni (nidificate all'interno della pianifica	Oggetto di riferimento, ad esempio «schedule»: {"ref":} «myScheduleId}

Campi Object Invocation	Description	Tipo di slot
	zione principale), gli utenti possono creare un oggetto padre che dispone di un riferimento alla pianificazione. Per ulteriori informazioni sulle configurazioni di pianificazione opzionali di esempio, consulta https://docs.aws.amazon.com/datapipeline/latest/DeveloperGuide/dp-object-schedule.html	
Campi opzionali	Description	Tipo di slot
attemptStatus	Lo stato segnalato più di recente dall'attività remota.	Stringa
attemptTimeout	Timeout per il completamento del lavoro in remoto. Se questo campo è impostato, un'attività remota che non viene completata entro il tempo impostato di avvio viene tentata di nuovo.	Periodo
compressione	Il tipo di compressione per i dati descritto da S3DataNode. «none» non è compressione e «gzip» viene compresso con l'algoritmo gzip. Questo campo è supportato solo per l'uso con Amazon Redshift e quando usi DataNode S3 con. CopyActivity	Enumerazione
dataFormat	DataFormat per i dati descritti da questo S3. DataNode	Oggetto di riferimento, ad esempio «dataFormat»: {"ref":» myDataFormat Id "}

Campi opzionali	Description	Tipo di slot
dependsOn	Specifica una dipendenza su un altro oggetto eseguibile	Oggetto di riferimento, ad esempio «dependsOn»: {"ref":» «} myActivityId
directoryPath	Percorso della directory Amazon S3 come URI: s3://my-bucket/. my-key-for-directory È necessario fornire un valore filePath o directoryPath.	Stringa
failureAndRerunMode	Descrive il comportamento del nodo consumer quando le dipendenze presentano un errore o vengono di nuovo eseguite	Enumerazione
filePath	Il percorso dell'oggetto in Amazon S3 come URI, ad esempio: s3://my-bucket/. my-key-for-file È necessario fornire un valore filePath o directoryPath. Questi rappresentano una cartella e un nome di file. Utilizzare il valore directoryPath per gestire più file in una directory.	Stringa
lateAfterTimeout	Il tempo trascorso dall'inizio della pipeline entro il quale l'oggetto deve essere completato. Viene attivato solo quando il tipo di pianificazione non è impostato su. ondemand	Periodo
manifestFilePath	Il percorso di Amazon S3 verso un file manifest nel formato supportato da Amazon Redshift. AWS Data Pipeline utilizza il file manifest per copiare i file Amazon S3 specificati nella tabella. Questo campo è valido solo quando un RedShiftCopyActivity fa riferimento a DataNode S3.	Stringa

Campi opzionali	Description	Tipo di slot
maxActiveInstances	Il numero massimo di istanze attive simultanee di un componente. Le riesecuzioni non contano ai fini del numero di istanze attive.	Numero intero
maximumRetries	Numero massimo di tentativi in caso di errore	Numero intero
onFail	Un'azione da eseguire quando l'oggetto corrente ha esito negativo.	Oggetto di riferimento, ad esempio «onFail»: {"ref":» myActionId «}
onLateAction	Azioni che devono essere attivate se un oggetto non è stato ancora pianificato o non è ancora completo.	Oggetto di riferimento, ad esempio "onLateAction«: {"ref":» myActionId «}
onSuccess	Un'operazione da eseguire quando l'oggetto corrente ha esito positivo.	Oggetto di riferimento, ad esempio «onSuccess»: {"ref":» myActionId «}
parent	Padre dell'oggetto corrente da cui saranno ereditati gli slot.	Oggetto di riferimento, ad esempio «parent»: {"ref":» myBaseObject Id "}
pipelineLogUri	L'URI S3 (come 's3://BucketName/Key/ ') per caricare i log per la pipeline.	Stringa
precondizione	Definisce eventualmente una precondizione. Un nodo dati non è contrassegnato come "READY" finché tutte le precondizioni non siano state soddisfatte.	Oggetto di riferimento, ad esempio «precondition»: {"ref":» «} myPreconditionId

Campi opzionali	Description	Tipo di slot
reportProgressTimeout	Timeout per chiamate successive di attività in remoto a reportProgress. Se impostato, le attività in remoto che non presentano avanzamenti nel periodo specificato potrebbero essere considerate bloccate e sono quindi oggetto di un altro tentativo.	Periodo
retryDelay	La durata del timeout tra due tentativi.	Periodo
runsOn	Le risorse di calcolo per eseguire l'attività o il comando. Ad esempio, un'istanza Amazon EC2 o un cluster Amazon EMR.	Oggetto di riferimento, ad esempio «runSon»: {"ref":» myResourceId «}
s3 EncryptionType	Ignora il tipo di crittografia di Amazon S3. I valori sono SERVER_SIDE_ENCRYPTION o NONE. La crittografia lato server è abilitata per impostazione predefinita.	Enumerazione

Campi opzionali	Description	Tipo di slot
scheduleType	<p>Il tipo di pianificazione consente di specificare se gli oggetti nella definizione di pipeline devono essere programmati all'inizio o alla fine dell'intervallo. Time Series Style Scheduling significa che le istanze vengono programmate al termine di ogni intervallo e Cron Style Scheduling significa che le istanze vengono programmate all'inizio di ogni intervallo. Una pianificazione on demand consente di eseguire una pipeline una sola volta, per attivazione. Questo significa che non è necessario clonare o ricreare la pipeline per eseguirla di nuovo. Se utilizzi una pianificazione on demand, devi specificarlo nell'oggetto predefinito e deve essere l'unico scheduleType specificato per gli oggetti della pipeline. Per utilizzare le pipeline su richiesta, è sufficiente chiamare l'ActivatePipeline operazione per ogni esecuzione e successiva. I valori sono: cron, ondemand e timeseries.</p>	Enumerazione
workerGroup	<p>Il gruppo di lavoro. Utilizzato per le attività di routing. Se si fornisce un valore runsOn e workerGroup esiste, workerGroup verrà ignorato.</p>	Stringa

Campi Runtime	Description	Tipo di slot
@activeInstances	<p>Elenco di oggetti di istanze attive attualmente programmate.</p>	<p>Oggetto di riferimento, ad esempio «activeInstances»: {"ref":» Id "} myRunnableObject</p>

Campi Runtime	Description	Tipo di slot
@actualEndTime	L'ora in cui è terminata l'esecuzione di questo oggetto.	DateTime
@actualStartTime	L'ora in cui è stata avviata l'esecuzione di questo oggetto.	DateTime
cancellationReason	CancellationReason se questo oggetto è stato annullato.	Stringa
@cascadeFailedOn	Descrizione della catena di dipendenza che ha generato l'errore dell'oggetto.	Oggetto di riferimento, ad esempio "cascadeFailedOn«: {" ref»:» myRunnableObject Id "}
emrStepLog	Log della fase EMR disponibili solo sui tentativi delle attività EMR	Stringa
errorId	ErrorId se l'oggetto non è riuscito.	Stringa
errorMessage	ErrorMessage se l'oggetto non è riuscito.	Stringa
errorStackTrace	Traccia dello stack di errore se l'oggetto non è riuscito.	Stringa
@finishedTime	L'ora in cui è terminata l'esecuzione di questo oggetto.	DateTime
hadoopJobLog	Log delle attività Hadoop disponibili per le attività basate su EMR.	Stringa
@healthStatus	Lo stato di integrità dell'oggetto che riflette l'esito positivo o negativo dell'ultima istanza dell'oggetto che ha raggiunto lo stato di un'istanza terminata.	Stringa

Campi Runtime	Description	Tipo di slot
@healthStatusFromInstanceId	Id dell'ultimo oggetto dell'istanza che ha raggiunto lo stato terminato.	Stringa
@healthStatusUpdatedOra	L'ora in cui lo stato di integrità è stato aggiornato o l'ultima volta.	DateTime
hostname	Il nome host del client che si è aggiudicato il tentativo dell'attività.	Stringa
@lastDeactivatedTime	L'ora in cui l'oggetto è stato disattivato.	DateTime
@latestCompletedRunOra	L'orario dell'esecuzione più recente durante il quale l'esecuzione è stata completata.	DateTime
@latestRunTime	L'orario dell'esecuzione più recente durante il quale l'esecuzione è stata pianificata.	DateTime
@nextRunTime	L'orario dell'esecuzione da programmare come successiva.	DateTime
reportProgressTime	Il periodo di tempo più recente in cui l'attività remota ha segnalato un progresso.	DateTime
@scheduledEndTime	L'orario di termine della pianificazione per un oggetto	DateTime
@scheduledStartTime	L'orario di inizio della pianificazione per l'oggetto	DateTime
@status	Lo stato di questo oggetto.	Stringa
@version	Versione della pipeline con cui l'oggetto è stato creato.	Stringa

Campi Runtime	Description	Tipo di slot
@waitingOn	Descrizione dell'elenco di dipendenze per cui questo oggetto è in attesa.	Oggetto di riferimento, ad esempio «waitingOn»: {"ref":» myRunnableObject Id "}
Campi di sistema	Description	Tipo di slot
@error	Errore che descrive il formato oggetto errato	Stringa
@pipelineId	L'id della pipeline a cui appartiene questo oggetto	Stringa
@sphere	La sfera di un oggetto indica la propria posizione nel ciclo di vita: i Component Objects generano Instance Objects che eseguono Attempt Objects	Stringa

Vedi anche

- [MySQLDataNode](#)

SqlDataNode

Definisce un nodo dati utilizzando SQL.

Esempio

Di seguito è illustrato un esempio di questo tipo di oggetto. Questo oggetto fa riferimento a due altri oggetti definiti nello stesso file di definizione della pipeline. CopyPeriod è un oggetto Schedule e Ready è un oggetto di preconditione.

```
{
  "id" : "Sql Table",
```

```

"type" : "SqlDataNode",
"schedule" : { "ref" : "CopyPeriod" },
"table" : "adEvents",
"database": "myDataBaseName",
"selectQuery" : "select * from #{table} where eventTime >=
'#{@scheduledStartTime.format('YYYY-MM-dd HH:mm:ss')}' and eventTime <
'#{@scheduledEndTime.format('YYYY-MM-dd HH:mm:ss')}'",
"precondition" : { "ref" : "Ready" }
}

```

Sintassi

Campi obbligatori	Description	Tipo di slot
table	Nome della tabella nel database SQL.	Stringa

Campi Object Invocation	Description	Tipo di slot
schedule	<p>Questo oggetto viene richiamato entro l'esecuzione di un intervallo di pianificazione. Gli utenti devono specificare un riferimento alla pianificazione di un altro oggetto per impostare l'ordine di esecuzione delle dipendenze per questo oggetto. Gli utenti possono soddisfare questo requisito impostando esplicitamente una pianificazione sull'oggetto, ad esempio specificando «schedule»: {"ref": "«"}. DefaultSchedule Nella maggior parte dei casi, è meglio inserire il riferimento alla pianificazione nell'oggetto pipeline di default, in modo che tutti gli oggetti possano ereditare tale pianificazione. O, se la pipeline consiste di una struttura di pianificazioni (nidificate all'interno della pianificazione principale), gli utenti possono creare un oggetto padre che dispone di un riferimento</p>	<p>Oggetto di riferimento, ad esempio «schedule»: {"ref": «myScheduleId»}</p>

Campi Object Invocation	Description	Tipo di slot
	<p>alla pianificazione. Per ulteriori informazioni sulle configurazioni di pianificazione opzionali di esempio, consulta https://docs.aws.amazon.com/datapipeline/latest/DeveloperGuide/dp-object-schedule.html</p>	
Campi opzionali	Description	Tipo di slot
attemptStatus	Lo stato segnalato più di recente dall'attività remota.	Stringa
attemptTimeout	Timeout per il completamento del lavoro in remoto. Se questo campo è impostato, un'attività remota che non viene completata entro il tempo impostato di avvio viene tentata di nuovo.	Periodo
createTableSql	Un'espressione SQL per creare la tabella che crea la tabella.	Stringa
database	Nome del database.	Oggetto di riferimento, ad esempio «database»: {"ref":» myDatabaseId «}
dependsOn	Specifica la dipendenza su un altro oggetto eseguibile.	Oggetto di riferimento, ad esempio «dependsOn»: {"ref":» myActivityId «}

Campi opzionali	Description	Tipo di slot
failureAndRerunModo	Descrive il comportamento del nodo consumer quando le dipendenze presentano un errore o vengono di nuovo eseguite.	Enumerazione
insertQuery	Un'istruzione SQL per inserire dati nella tabella.	Stringa
lateAfterTimeout	Il tempo trascorso dopo l'inizio della pipeline entro il quale l'oggetto deve essere completato. Viene attivato solo quando il tipo di pianificazione non è impostato su. ondemand	Periodo
maxActiveInstances	Il numero massimo di istanze attive simultanee di un componente. Le riesecuzioni non contano ai fini del numero di istanze attive.	Numero intero
maximumRetries	Numero massimo di tentativi in caso di errore	Numero intero
onFail	Un'azione da eseguire quando l'oggetto corrente ha esito negativo.	Oggetto di riferimento, ad esempio «onFail»: {"ref":» myActionId «}
onLateAction	Azioni che devono essere attivate se un oggetto non è stato ancora pianificato o non è ancora completo.	Oggetto di riferimento, ad esempio "onLateAction«: {"ref":» myActionId «}
onSuccess	Un'operazione da eseguire quando l'oggetto corrente ha esito positivo.	Oggetto di riferimento, ad esempio «onSuccess»: {"ref":» myActionId «}
parent	Padre dell'oggetto corrente da cui saranno ereditati gli slot.	Oggetto di riferimento, ad esempio «parent»: {"ref":» myBaseObject Id "}

Campi opzionali	Description	Tipo di slot
pipelineLogUri	L'URI S3 (come 's3://BucketName/Key/ ') per caricare i log per la pipeline.	Stringa
precondizione	Definisce eventualmente una precondizione. Un nodo dati non è contrassegnato come "READY" finché tutte le precondizioni non siano state soddisfatte.	Oggetto di riferimento, ad esempio «precondition»: {"ref":» «} myPreconditionId
reportProgressTimeout	Timeout per chiamate successive di attività in remoto a reportProgress. Se impostato, le attività in remoto che non presentano avanzamenti nel periodo specificato potrebbero essere considerate bloccate e sono quindi oggetto di un altro tentativo.	Periodo
retryDelay	La durata del timeout tra due tentativi.	Periodo
runsOn	Le risorse di calcolo per eseguire l'attività o il comando. Ad esempio, un'istanza Amazon EC2 o un cluster Amazon EMR.	Oggetto di riferimento, ad esempio «runson»: {"ref":» «} myResourceId «}

Campi opzionali	Description	Tipo di slot
scheduleType	Il tipo di pianificazione consente di specificare se gli oggetti nella definizione di pipeline devono essere programmati all'inizio o alla fine dell'intervallo. Time Series Style Scheduling significa che le istanze vengono programmate al termine di ogni intervallo e Cron Style Scheduling significa che le istanze vengono programmate all'inizio di ogni intervallo. Una pianificazione on demand consente di eseguire una pipeline una sola volta, per attivazione. Questo significa che non è necessario clonare o ricreare la pipeline per eseguirla di nuovo. Se utilizzi una pianificazione on demand, devi specificarlo nell'oggetto predefinito e deve essere l'unico scheduleType specificato per gli oggetti della pipeline. Per utilizzare le pipeline su richiesta, è sufficiente chiamare l'ActivatePipeline operazione per ogni esecuzione e successiva. I valori sono: cron, ondemand e timeseries.	Enumerazione
schemaName	Il nome dello schema in cui è presente la tabella.	Stringa
selectQuery	Un'istruzione SQL per recuperare i dati dalla tabella.	Stringa
workerGroup	Il gruppo di lavoro. Utilizzato per le attività di routing. Se si fornisce un valore runsOn e workerGroup esiste, workerGroup verrà ignorato.	Stringa

Campi Runtime	Description	Tipo di slot
@activeInstances	Elenco di oggetti di istanze attive attualmente programmate.	Oggetto di riferimento, ad esempio «activeInstances»: {"ref":» Id " } myRunnableObject
@actualEndTime	L'ora in cui è terminata l'esecuzione di questo oggetto.	DateTime
@actualStartTime	L'ora in cui è stata avviata l'esecuzione di questo oggetto.	DateTime
cancellationReason	CancellationReason se questo oggetto è stato annullato.	Stringa
@cascadeFailedOn	Descrizione della catena di dipendenza che ha generato l'errore dell'oggetto.	Oggetto di riferimento, ad esempio "cascadeFailedOn«: {" ref":» myRunnableObject Id " }
emrStepLog	Log della fase EMR disponibili solo sui tentativi delle attività EMR	Stringa
errorId	ErrorId se l'oggetto non è riuscito.	Stringa
errorMessage	ErrorMessage se l'oggetto non è riuscito.	Stringa
errorStackTrace	Traccia dello stack di errore se l'oggetto non è riuscito.	Stringa
@finishedTime	L'ora in cui è terminata l'esecuzione di questo oggetto.	DateTime
hadoopJobLog	Log delle attività Hadoop disponibili per le attività basate su EMR.	Stringa
@healthStatus	Lo stato di integrità dell'oggetto che riflette l'esito positivo o negativo dell'ultima istanza	Stringa

Campi Runtime	Description	Tipo di slot
	dell'oggetto che ha raggiunto lo stato di un'istanza terminata.	
@healthStatusFromInstanceId	Id dell'ultimo oggetto dell'istanza che ha raggiunto lo stato terminato.	Stringa
@healthStatusUpdated Ora	L'ora in cui lo stato di integrità è stato aggiornato o l'ultima volta.	DateTime
hostname	Il nome host del client che si è aggiudicato il tentativo dell'attività.	Stringa
@lastDeactivatedTime	L'ora in cui l'oggetto è stato disattivato.	DateTime
@latestCompletedRun Ora	L'orario dell'esecuzione più recente durante il quale l'esecuzione è stata completata.	DateTime
@latestRunTime	L'orario dell'esecuzione più recente durante il quale l'esecuzione è stata pianificata.	DateTime
@nextRunTime	L'orario dell'esecuzione da programmare come successiva.	DateTime
reportProgressTime	Il periodo di tempo più recente in cui l'attività remota ha segnalato un progresso.	DateTime
@scheduledEndTime	L'orario di termine della pianificazione per un oggetto	DateTime
@scheduledStartTime	L'orario di inizio della pianificazione per l'oggetto	DateTime
@status	Lo stato di questo oggetto.	Stringa
@version	Versione della pipeline con cui l'oggetto è stato creato.	Stringa

Campi Runtime	Description	Tipo di slot
@waitingOn	Descrizione dell'elenco di dipendenze per cui questo oggetto è in attesa.	Oggetto di riferimento, ad esempio «waitingOn»: {"ref":» myRunnableObject Id "}
Campi di sistema	Description	Tipo di slot
@error	Errore che descrive il formato oggetto errato	Stringa
@pipelineId	L'id della pipeline a cui appartiene questo oggetto	Stringa
@sphere	La sfera di un oggetto indica la propria posizione nel ciclo di vita: i Component Objects generano Instance Objects che eseguono Attempt Objects	Stringa

Vedi anche

- [S3 DataNode](#)

Attività

Di seguito sono elencati gli oggetti attività di AWS Data Pipeline:

Oggetti

- [CopyActivity](#)
- [EmrActivity](#)
- [HadoopActivity](#)
- [HiveActivity](#)
- [HiveCopyActivity](#)

- [PigActivity](#)
- [RedshiftCopyActivity](#)
- [ShellCommandActivity](#)
- [SqlActivity](#)

CopyActivity

Copia i dati da una posizione all'altra. CopyActivity supporta [S3 DataNode](#) e [SqlDataNode](#) come input e output e l'operazione di copia viene normalmente eseguita record-by-record. Tuttavia, CopyActivity fornisce una copia da Amazon S3 ad Amazon S3 ad alte prestazioni quando sono soddisfatte tutte le seguenti condizioni:

- L'input e l'output sono S3 DataNodes
- Il campo dataFormat è lo stesso per input e output

Se si forniscono file di dati compressi come input senza indicarlo utilizzando il campo compression sui nodi di dati S3, CopyActivity potrebbe non riuscire. In questo caso, CopyActivity non è in grado di rilevare correttamente il termine del carattere del record e l'operazione ha esito negativo. Inoltre, CopyActivity supporta la copia da una directory a un'altra directory e la copia di un file in una directory, ma la record-by-record copia si verifica quando si copia una directory in un file. Infine, non CopyActivity supporta la copia di file Amazon S3 multiparte.

CopyActivity ha limitazioni specifiche per il supporto CSV. Quando usi un S3 DataNode come input per CopyActivity, puoi usare solo una variante Unix/Linux del formato di file di dati CSV per i campi di input e output di Amazon S3. La variante Unix/Linux richiede quanto segue:

- Il separatore deve essere il carattere "," (virgola).
- I record non sono citati.
- Il carattere di escape predefinito è il valore ASCII 92 (barra rovesciata).
- La fine dell'identificatore del record è il valore ASCII 10 (o "\n").

I sistemi basati su Windows utilizzano in genere una sequenza di end-of-record caratteri diversa: un carriage return e un line feed insieme (valore ASCII 13 e valore ASCII 10). È necessario adeguarsi a questa differenza utilizzando un ulteriore meccanismo, come uno script di pre-copia per modificare

i dati di input, e assicurare che CopyActivity sia in grado di rilevare correttamente la fine di un record; in caso contrario, CopyActivity ha ripetutamente esito negativo.

Quando si utilizza CopyActivity per eseguire l'esportazione da un oggetto RDS PostgreSQL a un formato di dati TSV, il carattere NULL predefinito è \n.

Esempio

Di seguito è illustrato un esempio di questo tipo di oggetto. Questo oggetto fa riferimento a tre altri oggetti definiti nello stesso file di definizione della pipeline. CopyPeriod è un oggetto Schedule e InputData e OutputData sono oggetti di nodi di dati.

```
{
  "id" : "S3ToS3Copy",
  "type" : "CopyActivity",
  "schedule" : { "ref" : "CopyPeriod" },
  "input" : { "ref" : "InputData" },
  "output" : { "ref" : "OutputData" },
  "runsOn" : { "ref" : "MyEc2Resource" }
}
```

Sintassi

Campi Object Invocation	Description	Tipo di slot
schedule	Questo oggetto viene richiamato entro l'esecuzione di un intervallo di pianificazione. Gli utenti devono specificare un riferimento alla pianificazione di un altro oggetto per impostare l'ordine di esecuzione delle dipendenze per questo oggetto. Gli utenti possono soddisfare questo requisito impostando esplicitamente una pianificazione sull'oggetto, ad esempio specificando «schedule»: {"ref": "«»}. DefaultSchedule Nella maggior parte dei casi, è meglio inserire il riferimento alla pianificazione nell'oggetto pipeline di default, in modo che tutti gli oggetti possano ereditare tale pianificazione.	Oggetto di riferimento, ad esempio «schedule»: {"ref": "«myScheduleId»}

Campi Object Invocation	Description	Tipo di slot
	O, se la pipeline consiste di una struttura di pianificazioni (nidificate all'interno della pianificazione principale), gli utenti possono creare un oggetto padre che dispone di un riferimento alla pianificazione. Per ulteriori informazioni sulle configurazioni di pianificazione opzionali di esempio, consulta https://docs.aws.amazon.com/datapipeline/latest/DeveloperGuide/dp-object-schedule.html	

Gruppo richiesto (uno dei seguenti è obbligatorio)	Description	Tipo di slot
runsOn	Le risorse di calcolo per eseguire l'attività o il comando. Ad esempio, un'istanza Amazon EC2 o un cluster Amazon EMR.	Oggetto di riferimento, ad esempio «runSon»: {"ref":» myResourceId «}
workerGroup	Il gruppo di lavoro. Utilizzato per le attività di routing. Se si fornisce un valore runsOn e workerGroup esiste, workerGroup verrà ignorato.	Stringa

Campi opzionali	Description	Tipo di slot
attemptStatus	Lo stato segnalato più di recente dall'attività remota.	Stringa
attemptTimeout	Timeout per il completamento del lavoro in remoto. Se questo campo è impostato, un'attività remota che non viene completata entro	Periodo

Campi opzionali	Description	Tipo di slot
	il tempo impostato di avvio viene tentata di nuovo.	
dependsOn	Specifica una dipendenza su un altro oggetto eseguibile.	Oggetto di riferimento, ad esempio «dependsOn»: {"ref":» «myActivityId}
failureAndRerunModo	Descrive il comportamento del nodo consumer quando le dipendenze presentano un errore o vengono di nuovo eseguite	Enumerazione
input	Origine dati di input.	Oggetto di riferimento, ad esempio «input»: {"ref":» myDataNodeId "}
lateAfterTimeout	Il tempo trascorso dall'inizio della pipeline entro il quale l'oggetto deve essere completato. Viene attivato solo quando il tipo di pianificazione non è impostato su. ondemand	Periodo
maxActiveInstances	Il numero massimo di istanze attive simultanee di un componente. Le riesecuzioni non contano ai fini del numero di istanze attive.	Numero intero
maximumRetries	Numero massimo di tentativi in caso di errore	Numero intero
onFail	Un'azione da eseguire quando l'oggetto corrente ha esito negativo.	Oggetto di riferimento, ad esempio «onFail»: {"ref":» myActionId «}
onLateAction	Azioni che devono essere attivate se un oggetto non è stato ancora pianificato o non è ancora completo.	Oggetto di riferimento, ad esempio "onLateAction«: {"ref":» myActionId «}

Campi opzionali	Description	Tipo di slot
onSuccess	Un'operazione da eseguire quando l'oggetto corrente ha esito positivo.	Oggetto di riferimento, ad esempio «onSuccess»: {"ref":» myActionId «}
output	Origine dati di output.	Oggetto di riferimento, ad esempio «output»: {"ref":» myDataNodeId "}
parent	Padre dell'oggetto corrente da cui saranno ereditati gli slot.	Oggetto di riferimento, ad esempio «parent»: {"ref":» myBaseObject Id "}
pipelineLogUri	L'URI S3 (come 's3://BucketName/Key/ ') per caricare i log per la pipeline.	Stringa
precondizione	Definisce eventualmente una precondizione. Un nodo dati non è contrassegnato come "READY" finché tutte le precondizioni non siano state soddisfatte.	Oggetto di riferimento, ad esempio «precondition»: {"ref":» «} myPreconditionId
reportProgressTimeout	Timeout per chiamate successive di attività in remoto a reportProgress. Se impostato, le attività in remoto che non presentano avanzamenti nel periodo specificato potrebbero essere considerate bloccate e sono quindi oggetto di un altro tentativo.	Periodo
retryDelay	La durata del timeout tra due tentativi.	Periodo

Campi opzionali	Description	Tipo di slot
scheduleType	<p>Il tipo di pianificazione consente di specificare se gli oggetti nella definizione di pipeline devono essere programmati all'inizio o alla fine dell'intervallo. Time Series Style Scheduling significa che le istanze vengono programmate al termine di ogni intervallo e Cron Style Scheduling significa che le istanze vengono programmate all'inizio di ogni intervallo. Una pianificazione on demand consente di eseguire una pipeline una sola volta, per attivazione. Questo significa che non è necessario clonare o ricreare la pipeline per eseguirla di nuovo. Se utilizzi una pianificazione on demand, devi specificarlo nell'oggetto predefinito e deve essere l'unico scheduleType specificato per gli oggetti della pipeline. Per utilizzare le pipeline su richiesta, è sufficiente chiamare l'ActivatePipeline operazione per ogni esecuzione e successiva. I valori sono: cron, ondemand e timeseries.</p>	Enumerazione

Campi Runtime	Description	Tipo di slot
@activeInstances	Elenco di oggetti di istanze attive attualmente programmate.	Oggetto di riferimento, ad esempio «activeInstances»: {"ref":» Id "}" myRunnableObject
@actualEndTime	L'ora in cui è terminata l'esecuzione di questo oggetto.	DateTime
@actualStartTime	L'ora in cui è stata avviata l'esecuzione di questo oggetto.	DateTime

Campi Runtime	Description	Tipo di slot
cancellationReason	CancellationReason se questo oggetto è stato annullato.	Stringa
@cascadeFailedOn	Descrizione della catena di dipendenza che ha generato l'errore dell'oggetto.	Oggetto di riferimento, ad esempio "cascadeFailedOn«: {" ref»:» myRunnableObject Id "}
emrStepLog	Log della fase EMR disponibili solo sui tentativi delle attività EMR	Stringa
errorId	ErrorId se l'oggetto non è riuscito.	Stringa
errorMessage	ErrorMessage se l'oggetto non è riuscito.	Stringa
errorStackTrace	Traccia dello stack di errore se l'oggetto non è riuscito.	Stringa
@finishedTime	L'ora in cui è terminata l'esecuzione di questo oggetto.	DateTime
hadoopJobLog	Log delle attività Hadoop disponibili per le attività basate su EMR.	Stringa
@healthStatus	Lo stato di integrità dell'oggetto che riflette l'esito positivo o negativo dell'ultima istanza dell'oggetto che ha raggiunto lo stato di un'istanza terminata.	Stringa
@healthStatusFromInstanceid	Id dell'ultimo oggetto dell'istanza che ha raggiunto lo stato terminato.	Stringa
@healthStatusUpdated Ora	L'ora in cui lo stato di integrità è stato aggiornato o l'ultima volta.	DateTime

Campi Runtime	Description	Tipo di slot
hostname	Il nome host del client che si è aggiudicato il tentativo dell'attività.	Stringa
@lastDeactivatedTime	L'ora in cui l'oggetto è stato disattivato.	DateTime
@latestCompletedRun Ora	L'orario dell'esecuzione più recente durante il quale l'esecuzione è stata completata.	DateTime
@latestRunTime	L'orario dell'esecuzione più recente durante il quale l'esecuzione è stata pianificata.	DateTime
@nextRunTime	L'orario dell'esecuzione da programmare come successiva.	DateTime
reportProgressTime	Il periodo di tempo più recente in cui l'attività remota ha segnalato un progresso.	DateTime
@scheduledEndTime	L'orario di termine della pianificazione per un oggetto	DateTime
@scheduledStartTime	L'orario di inizio della pianificazione per l'oggetto	DateTime
@status	Lo stato di questo oggetto.	Stringa
@version	Versione della pipeline con cui l'oggetto è stato creato.	Stringa
@waitingOn	Descrizione dell'elenco di dipendenze per cui questo oggetto è in attesa.	Oggetto di riferimento, ad esempio «waitingOn»: {"ref":» myRunnableObject Id "}

Campi di sistema	Description	Tipo di slot
@error	Errore che descrive il formato oggetto errato	Stringa
@pipelineId	L'id della pipeline a cui appartiene questo oggetto	Stringa
@sphere	La sfera di un oggetto indica la propria posizione nel ciclo di vita: i Component Objects generano Instance Objects che eseguono Attempt Objects	Stringa

Vedi anche

- [ShellCommandActivity](#)
- [EmrActivity](#)
- [Esporta dati MySQL su Amazon S3 utilizzando AWS Data Pipeline](#)

EmrActivity

Esegue un cluster EMR.

AWS Data Pipeline utilizza un formato diverso per i passaggi rispetto ad Amazon EMR; ad esempio, AWS Data Pipeline utilizza argomenti separati da virgole dopo il nome JAR nel campo step.

EmrActivity L'esempio seguente mostra un passaggio formattato per Amazon EMR, seguito dal AWS Data Pipeline suo equivalente:

```
s3://example-bucket/MyWork.jar arg1 arg2 arg3
```

```
"s3://example-bucket/MyWork.jar, arg1, arg2, arg3"
```

Esempi

Di seguito è illustrato un esempio di questo tipo di oggetto. Questo esempio utilizza versioni precedenti di Amazon EMR. Verifica la correttezza di questo esempio con la versione del cluster Amazon EMR che stai utilizzando.

Questo oggetto fa riferimento a tre altri oggetti definiti nello stesso file di definizione della pipeline. `MyEmrCluster` è un oggetto `EmrCluster` e `MyS3Input` e `MyS3Output` sono oggetti `S3DataNode`.

Note

In questo esempio, è possibile sostituire il campo `step` con la stringa di cluster desiderato, che può essere uno script Pig, un cluster di streaming Hadoop, un JAR personalizzato, inclusi i parametri, e così via.

Hadoop 2.x (AMI 3.x)

```
{
  "id" : "MyEmrActivity",
  "type" : "EmrActivity",
  "runsOn" : { "ref" : "MyEmrCluster" },
  "preStepCommand" : "scp remoteFiles localFiles",
  "step" : ["s3://mybucket/myPath/myStep.jar,firstArg,secondArg,-files,s3://mybucket/myPath/myFile.py,-input,s3://myinputbucket/path,-output,s3://myoutputbucket/path,-mapper,myFile.py,-reducer,reducerName","s3://mybucket/myPath/myotherStep.jar,..."],
  "postStepCommand" : "scp localFiles remoteFiles",
  "input" : { "ref" : "MyS3Input" },
  "output" : { "ref" : "MyS3Output" }
}
```

Note

Per passare argomenti a un'applicazione in una fase, è necessario specificare la regione nel percorso dello script, come nell'esempio seguente. Inoltre, potrebbe essere necessario uscire dagli argomenti passati. Ad esempio, se si utilizza `script-runner.jar` per eseguire uno script shell e si desidera passare argomenti allo script, è necessario eliminare le virgole che li separano. Lo slot della fase che segue dimostra come eseguire questa operazione:

```
"step" : "s3://eu-west-1.elasticmapreduce/libs/script-runner/script-runner.jar,s3://datapipeline/echo.sh,a\\,\\,b\\,\\,c"
```

In questa fase viene utilizzato `script-runner.jar` per eseguire lo `echo.sh` script shell e passare `a`, `b` e `c` come unico argomento allo script. Il primo carattere escape viene rimosso

dall'argomento risultante in modo da poterlo utilizzare di nuovo. Ad esempio, se si ha `File \.gz` come argomento in formato JSON, è possibile farlo usando `File\\\\.gz`. Tuttavia, poiché il primo carattere escape viene eliminato, è necessario utilizzare `File\\\\\\\\.gz`.

Sintassi

Campi Object Invocation	Description	Tipo di slot
schedule	<p>Questo oggetto viene richiamato entro l'esecuzione di un intervallo di pianificazione. Specificare un riferimento alla pianificazione di un altro oggetto per impostare l'ordine di esecuzione e delle dipendenze per questo oggetto. È possibile soddisfare questo requisito impostando esplicitamente una pianificazione sull'oggetto, ad esempio, specificando <code>"schedule": {"ref": "DefaultSchedule"}</code>.</p> <p>Nella maggior parte dei casi, è meglio inserire il riferimento alla pianificazione nell'oggetto pipeline di default, in modo che tutti gli oggetti possano ereditare tale pianificazione. O, se la pipeline consiste di una struttura di pianificazioni (nidificate all'interno della pianificazione principale), è possibile creare un oggetto padre che dispone di un riferimento alla pianificazione. Per ulteriori informazioni sulle configurazioni di pianificazione opzionali di esempio, consulta https://docs.aws.amazon.com/datapipeline/latest/DeveloperGuide/dp-object-schedule.html</p>	<p>Oggetto di riferimento, ad esempio <code>«schedule»: {"ref":» myScheduleId «}</code></p>

Gruppo richiesto (uno dei seguenti è obbligatorio)	Description	Tipo di slot
runsOn	Il cluster Amazon EMR su cui verrà eseguito questo processo.	Oggetto di riferimento, ad esempio «runsOn»: {"ref":» myEmrCluster Id "}
workerGroup	Il gruppo di lavoro. Utilizzato per le attività di routing. Se si fornisce un valore runsOn ed esiste workerGroup , workerGroup verrà ignorato.	Stringa

Campi opzionali	Description	Tipo di slot
attemptStatus	Lo stato segnalato più di recente dall'attività remota.	Stringa
attemptTimeout	Timeout per il completamento del lavoro in remoto. Se questo campo è impostato, un'attività remota che non viene completata entro il tempo impostato di avvio viene tentata di nuovo.	Periodo
dependsOn	Specifica una dipendenza su un altro oggetto eseguibile.	Oggetto di riferimento, ad esempio «dependsOn»: {"ref":» myActivityId «}
failureAndRerunModo	Descrive il comportamento del nodo consumer quando le dipendenze presentano un errore o vengono di nuovo eseguite.	Enumerazione

Campi opzionali	Description	Tipo di slot
input	Posizione dei dati di input.	Oggetto di riferimento, ad esempio, «input»: {"ref":» myDataNode Id "}
lateAfterTimeout	Il tempo trascorso dall'inizio della pipeline entro il quale l'oggetto deve essere completato. Viene attivato solo quando il tipo di pianificazione non è impostato su. ondemand	Periodo
maxActiveInstances	Il numero massimo di istanze attive simultanee di un componente. Le riesecuzioni non contano ai fini del numero di istanze attive.	Numero intero
maximumRetries	Numero massimo di tentativi in caso di errore.	Numero intero
onFail	Un'azione da eseguire quando l'oggetto corrente ha esito negativo.	Oggetto di riferimento, ad esempio «onFail»: {"ref":» myActionId «}
onLateAction	Azioni che devono essere attivate se un oggetto non è stato ancora pianificato o non è ancora completo.	Oggetto di riferimento, ad esempio "onLateAction«: {"ref":» myActionId «}
onSuccess	Un'operazione da eseguire quando l'oggetto corrente ha esito positivo.	Oggetto di riferimento, ad esempio «onSuccess»: {"ref":» myActionId «}
output	Posizione dei dati di output.	Oggetto di riferimento, ad esempio «output»: {"ref":» myDataNode Id "}

Campi opzionali	Description	Tipo di slot
parent	Padre dell'oggetto corrente da cui saranno ereditati gli slot.	Oggetto di riferimento, ad esempio «parent»: {"ref":» myBaseObject Id "}
pipelineLogUri	L'URI di Amazon S3, ad esempio 's3://BucketName/Prefix/ 'per caricare i log per la pipeline.	Stringa
postStepCommand	Script di shell da eseguire dopo il completamento di tutti i passaggi. Per specificare più script, fino a 255, aggiungere più campi postStepCommand .	Stringa
precondizione	Definisce eventualmente una precondizione. Un nodo dati non è contrassegnato come "READY" finché tutte le precondizioni non siano state soddisfatte.	Oggetto di riferimento, ad esempio, «precondition»: {"ref":» «} myPreconditionId
preStepCommand	Script di shell da eseguire prima dell'esecuzione di qualsiasi passaggio. Per specificare più script, fino a 255, aggiungere più campi preStepCommand .	Stringa
reportProgressTimeout	Timeout per chiamate successive di attività in remoto a reportProgress . Se impostato , le attività in remoto che non presentano avanzamenti nel periodo specificato potrebbero essere considerate bloccate e sono quindi oggetto di un altro tentativo.	Periodo

Campi opzionali	Description	Tipo di slot
<code>resizeClusterBeforeInEsecuzione</code>	Ridimensiona il cluster prima di eseguire questa attività per adattarlo alle tabelle DynamoDB specificate come input o output. <div data-bbox="472 401 1149 1003"><p> Note</p><p>Se <code>EmrActivity</code> utilizza una <code>DynamoDBDataNode</code> come nodo di dati di input o output e se imposti <code>resizeClusterBeforeRunning</code> su <code>TRUE</code>, AWS Data Pipeline viene avviata utilizzando i tipi di istanza <code>m3.xlarge</code>. Questo sovrascrive le tue scelte in termini di tipi di istanze con <code>m3.xlarge</code>, con un possibile aumento dei costi.</p></div>	Booleano
<code>resizeClusterMaxIstanze</code>	Un limite per il numero massimo di istanze che possono essere richieste dall'algoritmo di ridimensionamento.	Numero intero
<code>retryDelay</code>	La durata del timeout tra due tentativi.	Periodo

Campi opzionali	Description	Tipo di slot
<code>scheduleType</code>	Il tipo di pianificazione consente di specificare se gli oggetti nella definizione di pipeline devono essere programmati all'inizio o alla fine dell'intervallo. I valori sono <code>cron</code> , <code>ondemand</code> e <code>timeseries</code> . La pianificazione <code>timeseries</code> significa che le istanze sono programmate al termine di ogni intervallo. La pianificazione <code>cron</code> significa che le istanze sono programmate all'inizio di ogni intervallo. Una pianificazione <code>ondemand</code> consente di eseguire una pipeline una sola volta, per attivazione. Non è necessario clonare o ricreare la pipeline per eseguirla di nuovo. Se utilizzi una pianificazione <code>ondemand</code> , devi specificarlo nell'oggetto predefinito e deve essere l'unico <code>scheduleType</code> specificato per gli oggetti della pipeline. Per utilizzare le pipeline <code>ondemand</code> , chiama l'operazione <code>ActivatePipeline</code> per ogni esecuzione successiva.	Enumerazione
<code>fase</code>	Uno o più passaggi per il cluster da eseguire. Per specificare più passaggi, fino a 255, aggiungere più campi relativi a queste informazioni. Utilizza argomenti separati da virgole dopo il nome JAR; ad esempio, <code>"s3://example-bucket/MyWork.jar, arg1, arg2, arg3"</code> .	Stringa

Campi Runtime	Description	Tipo di slot
<code>@activeInstances</code>	Elenco di oggetti di istanze attive attualmente programmate.	Oggetto di riferimento, ad esempio «activeIn

Campi Runtime	Description	Tipo di slot
		stances»: {"ref»:» Id "} myRunnableObject
@actualEndTime	L'ora in cui è terminata l'esecuzione di questo oggetto.	DateTime
@actualStartTime	L'ora in cui è stata avviata l'esecuzione di questo oggetto.	DateTime
cancellationReason	CancellationReason se questo oggetto è stato annullato.	Stringa
@cascadeFailedOn	Descrizione della catena di dipendenza che ha generato l'errore dell'oggetto.	Oggetto di riferimento, ad esempio "cascadeFailedOn«: {" ref»:» myRunnableObject Id "}
emrStepLog	I log dei passaggi di Amazon EMR sono disponibili solo nei tentativi di attività EMR	Stringa
errorId	errorId se l'oggetto non riuscito.	Stringa
errorMessage	errorMessage se l'oggetto non riuscito.	Stringa
errorStackTrace	Traccia dello stack di errore se l'oggetto non è riuscito.	Stringa
@finishedTime	L'ora in cui è terminata l'esecuzione di questo oggetto.	DateTime
hadoopJobLog	Log delle attività Hadoop disponibili per le attività basate su EMR.	Stringa
@healthStatus	Lo stato di integrità dell'oggetto che riflette l'esito positivo o negativo dell'ultima istanza dell'oggetto che ha raggiunto lo stato di un'istanza terminata.	Stringa

Campi Runtime	Description	Tipo di slot
@healthStatusFromInstanceid	Id dell'ultimo oggetto dell'istanza che ha raggiunto lo stato terminato.	Stringa
@ Ora healthStatusUpdated	L'ora in cui lo stato di integrità è stato aggiornato o l'ultima volta.	DateTime
hostname	Il nome host del client che si è aggiudicato il tentativo dell'attività.	Stringa
@lastDeactivatedTime	L'ora in cui l'oggetto è stato disattivato.	DateTime
@ latestCompletedRun Ora	L'orario dell'esecuzione più recente durante il quale l'esecuzione è stata completata.	DateTime
@latestRunTime	L'orario dell'esecuzione più recente durante il quale l'esecuzione è stata pianificata.	DateTime
@nextRunTime	L'orario dell'esecuzione da programmare come successiva.	DateTime
reportProgressTime	Il periodo di tempo più recente in cui l'attività remota ha segnalato un progresso.	DateTime
@scheduledEndTime	L'orario di termine della pianificazione per l'oggetto.	DateTime
@scheduledStartTime	L'orario di inizio della pianificazione per l'oggetto.	DateTime
@status	Lo stato di questo oggetto.	Stringa
@version	Versione della pipeline con cui l'oggetto è stato creato.	Stringa

Campi Runtime	Description	Tipo di slot
@waitingOn	Descrizione dell'elenco di dipendenze per cui questo oggetto è in attesa.	Oggetto di riferimento, ad esempio «waitingOn»: {"ref":» myRunnableObject Id "}
Campi di sistema	Description	Tipo di slot
@error	Errore che descrive il formato oggetto errato.	Stringa
@pipelineId	L'ID della pipeline a cui appartiene questo oggetto.	Stringa
@sphere	La sfera di un oggetto indica la propria posizione nel ciclo di vita: i Component Objects generano Instance Objects che eseguono Attempt Objects.	Stringa

Vedi anche

- [ShellCommandActivity](#)
- [CopyActivity](#)
- [EmrCluster](#)

HadoopActivity

Esegue un MapReduce processo su un cluster. Il cluster può essere un cluster EMR gestito da AWS Data Pipeline o un'altra risorsa, se si utilizza. TaskRunner Da utilizzare HadoopActivity quando si desidera eseguire il lavoro in parallelo. Ciò consente di utilizzare le risorse di pianificazione del framework YARN o del negoziatore di MapReduce risorse in Hadoop 1. Se desideri eseguire il lavoro in sequenza utilizzando l'azione Amazon EMR Step, puoi comunque utilizzare. [EmrActivity](#)

Esempi

HadoopActivity utilizzando un cluster EMR gestito da AWS Data Pipeline

L' HadoopActivity oggetto seguente utilizza una EmrCluster risorsa per eseguire un programma:

```
{
  "name": "MyHadoopActivity",
  "schedule": {"ref": "ResourcePeriod"},
  "runsOn": {"ref": "MyEmrCluster"},
  "type": "HadoopActivity",
  "preActivityTaskConfig": {"ref": "preTaskScriptConfig"},
  "jarUri": "/home/hadoop/contrib/streaming/hadoop-streaming.jar",
  "argument": [
    "-files",
    "s3://elasticmapreduce/samples/wordcount/wordSplitter.py",
    "-mapper",
    "wordSplitter.py",
    "-reducer",
    "aggregate",
    "-input",
    "s3://elasticmapreduce/samples/wordcount/input/",
    "-output",
    "s3://test-bucket/MyHadoopActivity/#{@pipelineId}/
    #{format(@scheduledStartTime, 'YYYY-MM-dd')}"
  ],
  "maximumRetries": "0",
  "postActivityTaskConfig": {"ref": "postTaskScriptConfig"},
  "hadoopQueue" : "high"
}
```

Ecco il corrispondente *MyEmrCluster*, che configura le code FairScheduler and in YARN per le AMI basate su Hadoop 2:

```
{
  "id" : "MyEmrCluster",
  "type" : "EmrCluster",
  "hadoopSchedulerType" : "PARALLEL_FAIR_SCHEDULING",
  "amiVersion" : "3.7.0",
  "bootstrapAction" : ["s3://Region.elasticmapreduce/bootstrap-
actions/configure-hadoop, -z, yarn.scheduler.capacity.root.queues=low
\,high\, default, -z, yarn.scheduler.capacity.root.high.capacity=50, -
```

```
z,yarn.scheduler.capacity.root.low.capacity=10,-
z,yarn.scheduler.capacity.root.default.capacity=30"]
}
```

Questo è quello che usi per configurare in Hadoop EmrCluster 1: FairScheduler

```
{
  "id": "MyEmrCluster",
  "type": "EmrCluster",
  "hadoopSchedulerType": "PARALLEL_FAIR_SCHEDULING",
  "amiVersion": "2.4.8",
  "bootstrapAction": "s3://Region.elasticmapreduce/bootstrap-
actions/configure-hadoop,-m,mapred.queue.names=low\\\\\\\\,high\\\\\\\\,default,-
m,mapred.fairscheduler.poolnameproperty=mapred.job.queue.name"
}
```

Le seguenti EmrCluster configurazioni CapacityScheduler per le AMI basate su Hadoop 2:

```
{
  "id": "MyEmrCluster",
  "type": "EmrCluster",
  "hadoopSchedulerType": "PARALLEL_CAPACITY_SCHEDULING",
  "amiVersion": "3.7.0",
  "bootstrapAction": "s3://Region.elasticmapreduce/bootstrap-
actions/configure-hadoop,-z,yarn.scheduler.capacity.root.queues=low
\\\\\\\\,high,-z,yarn.scheduler.capacity.root.high.capacity=40,-
z,yarn.scheduler.capacity.root.low.capacity=60"
}
```

HadoopActivity utilizzo di un cluster EMR esistente

In questo esempio, si utilizzano workergroups e TaskRunner a per eseguire un programma su un cluster EMR esistente. La seguente definizione di pipeline consente di: HadoopActivity

- Esegue un MapReduce programma solo sulle *myWorkerGroup* risorse. Per ulteriori informazioni sui gruppi di lavoratori, consulta [Esecuzione di lavori su risorse esistenti utilizzando Task Runner](#).
- Esegui un preActivityTask Config e Config postActivityTask

```
{
  "objects": [
    {
```

```

    "argument": [
      "-files",
      "s3://elasticmapreduce/samples/wordcount/wordSplitter.py",
      "-mapper",
      "wordSplitter.py",
      "-reducer",
      "aggregate",
      "-input",
      "s3://elasticmapreduce/samples/wordcount/input/",
      "-output",
      "s3://test-bucket/MyHadoopActivity/#{@pipelineId}/
#{format(@scheduledStartTime, 'YYYY-MM-dd')}"
    ],
    "id": "MyHadoopActivity",
    "jarUri": "/home/hadoop/contrib/streaming/hadoop-streaming.jar",
    "name": "MyHadoopActivity",
    "type": "HadoopActivity"
  },
  {
    "id": "SchedulePeriod",
    "startDateTime": "start_datettime",
    "name": "SchedulePeriod",
    "period": "1 day",
    "type": "Schedule",
    "endDateTime": "end_datettime"
  },
  {
    "id": "ShellScriptConfig",
    "scriptUri": "s3://test-bucket/scripts/preTaskScript.sh",
    "name": "preTaskScriptConfig",
    "scriptArgument": [
      "test",
      "argument"
    ],
    "type": "ShellScriptConfig"
  },
  {
    "id": "ShellScriptConfig",
    "scriptUri": "s3://test-bucket/scripts/postTaskScript.sh",
    "name": "postTaskScriptConfig",
    "scriptArgument": [
      "test",
      "argument"
    ],
  },

```

```

    "type": "ShellScriptConfig"
  },
  {
    "id": "Default",
    "scheduleType": "cron",
    "schedule": {
      "ref": "SchedulePeriod"
    },
    "name": "Default",
    "pipelineLogUri": "s3://test-bucket/logs/2015-05-22T18:02:00.343Z642f3fe415",
    "maximumRetries": "0",
    "workerGroup": "myWorkerGroup",
    "preActivityTaskConfig": {
      "ref": "preTaskScriptConfig"
    },
    "postActivityTaskConfig": {
      "ref": "postTaskScriptConfig"
    }
  }
]
}

```

Sintassi

Campi obbligatori	Description	Tipo di slot
jarUri	Posizione di un JAR in Amazon S3 o nel file system locale del cluster con cui eseguirlo. HadoopActivity	Stringa

Campi Object Invocation	Description	Tipo di slot
schedule	Questo oggetto viene richiamato entro l'esecuzione di un intervallo di pianificazione. Gli utenti devono specificare un riferimento alla pianificazione di un altro oggetto per impostare l'ordine di esecuzione delle dipendenze per	Oggetto di riferimento, ad esempio «schedule»: {"ref":» «myScheduleId}

Campi Object Invocation	Description	Tipo di slot
	<p>questo oggetto. Gli utenti possono soddisfare questo requisito impostando esplicitamente una pianificazione sull'oggetto, ad esempio specificando «schedule»: {"ref": "DefaultSchedule"}. Nella maggior parte dei casi, è meglio inserire il riferimento alla pianificazione nell'oggetto pipeline di default, in modo che tutti gli oggetti possano ereditare tale pianificazione. O, se la pipeline consiste di una struttura di pianificazioni (nidificate all'interno della pianificazione principale), gli utenti possono creare un oggetto padre che dispone di un riferimento alla pianificazione. Per ulteriori informazioni sulle configurazioni di pianificazione opzionali di esempio, consulta https://docs.aws.amazon.com/datapipeline/latest/DeveloperGuide/dp-object-schedule.html</p>	
Gruppo richiesto (uno dei seguenti è obbligatorio)	Description	Tipo di slot
runsOn	Cluster EMR su cui il processo verrà eseguito.	Oggetto di riferimento, ad esempio «runOn»: {"ref": "myEmrClusterId"}
workerGroup	Il gruppo di lavoro. Utilizzato per le attività di routing. Se si fornisce un valore runsOn e workerGroup esiste, workerGroup verrà ignorato.	Stringa

Campi opzionali	Description	Tipo di slot
argument	Gli argomenti da trasmettere al JAR.	Stringa
attemptStatus	Lo stato segnalato più di recente dall'attività remota.	Stringa
attemptTimeout	Timeout per il completamento del lavoro in remoto. Se questo campo è impostato, un'attività remota che non viene completata entro il tempo impostato di avvio viene tentata di nuovo.	Periodo
dependsOn	Specifica una dipendenza su un altro oggetto eseguibile.	Oggetto di riferimento, ad esempio «dependsOn»: {"ref":» «myActivityId}
failureAndRerunModo	Descrive il comportamento del nodo consumer quando le dipendenze presentano un errore o vengono di nuovo eseguite	Enumerazione
hadoopQueue	Il nome della coda del pianificatore Hadoop a cui verrà inviata l'attività.	Stringa
input	Posizione dei dati di input.	Oggetto di riferimento, ad esempio «input»: {"ref":» myDataNodeId "}
lateAfterTimeout	Il tempo trascorso dall'inizio della pipeline entro il quale l'oggetto deve essere completato. Viene attivato solo quando il tipo di pianificazione non è impostato su. ondemand	Periodo
mainClass	La classe principale del JAR con HadoopActivity cui stai eseguendo.	Stringa

Campi opzionali	Description	Tipo di slot
maxActiveInstances	Il numero massimo di istanze attive simultanee di un componente. Le riesecuzioni non contano ai fini del numero di istanze attive.	Numero intero
maximumRetries	Numero massimo di tentativi in caso di errore	Numero intero
onFail	Un'azione da eseguire quando l'oggetto corrente ha esito negativo.	Oggetto di riferimento, ad esempio «onFail»: {"ref":» myActionId «}
onLateAction	Azioni che devono essere attivate se un oggetto non è stato ancora pianificato o non è ancora completo.	Oggetto di riferimento, ad esempio "onLateAction«: {"ref":» myActionId «}
onSuccess	Un'operazione da eseguire quando l'oggetto corrente ha esito positivo.	Oggetto di riferimento, ad esempio «onSuccess»: {"ref":» myActionId «}
output	Posizione dei dati di output.	Oggetto di riferimento, ad esempio «output»: {"ref":» myDataNodeId "}
parent	Padre dell'oggetto corrente da cui saranno ereditati gli slot.	Oggetto di riferimento, ad esempio «parent»: {"ref":» myBaseObjectId "}
pipelineLogUri	L'URI S3 (come 's3://BucketName/Key/ ') per caricare i log per la pipeline.	Stringa

Campi opzionali	Description	Tipo di slot
postActivityTaskConfig	Lo script di configurazione post-attività da eseguire. Questo è composto da un URI dello script della shell in Amazon S3 e da un elenco di argomenti.	Oggetto di riferimento, ad esempio "postActivityTaskConfig": {"ref":» myShellScriptConfigId «}
preActivityTaskConfig	Lo script di configurazione pre-attività da eseguire. Questo è composto da un URI dello script della shell in Amazon S3 e da un elenco di argomenti.	Oggetto di riferimento, ad esempio "preActivityTaskConfig": {"ref":» myShellScriptConfigId «}
precondizione	Definisce eventualmente una precondizione. Un nodo dati non è contrassegnato come "READY" finché tutte le precondizioni non siano state soddisfatte.	Oggetto di riferimento, ad esempio «precondizione»: {"ref":» «myPreconditionId}
reportProgressTimeout	Timeout per chiamate successive di attività in remoto a reportProgress. Se impostato, le attività in remoto che non presentano avanzamenti nel periodo specificato potrebbero essere considerate bloccate e sono quindi oggetto di un altro tentativo.	Periodo
retryDelay	La durata del timeout tra due tentativi.	Periodo

Campi opzionali	Description	Tipo di slot
scheduleType	<p>Il tipo di pianificazione consente di specificare se gli oggetti nella definizione di pipeline devono essere programmati all'inizio o alla fine dell'intervallo. Time Series Style Scheduling significa che le istanze vengono programmate al termine di ogni intervallo e Cron Style Scheduling significa che le istanze vengono programmate all'inizio di ogni intervallo. Una pianificazione on demand consente di eseguire una pipeline una sola volta, per attivazione. Questo significa che non è necessario clonare o ricreare la pipeline per eseguirla di nuovo. Se utilizzi una pianificazione on demand, devi specificarlo nell'oggetto predefinito e deve essere l'unico scheduleType specificato per gli oggetti della pipeline. Per utilizzare le pipeline su richiesta, è sufficiente chiamare l'ActivatePipeline operazione per ogni esecuzione e successiva. I valori sono: cron, ondemand e timeseries.</p>	Enumerazione
Campi Runtime	Description	Tipo di slot
@activeInstances	Elenco di oggetti di istanze attive attualmente programmate.	Oggetto di riferimento, ad esempio «activeInstances»: {"ref":» Id "}" myRunnableObject
@actualEndTime	L'ora in cui è terminata l'esecuzione di questo oggetto.	DateTime
@actualStartTime	L'ora in cui è stata avviata l'esecuzione di questo oggetto.	DateTime

Campi Runtime	Description	Tipo di slot
cancellationReason	CancellationReason se questo oggetto è stato annullato.	Stringa
@cascadeFailedOn	Descrizione della catena di dipendenza che ha generato l'errore dell'oggetto.	Oggetto di riferimento, ad esempio "cascadeFailedOn«: {" ref»:» myRunnableObject Id "}
emrStepLog	Log della fase EMR disponibili solo sui tentativi delle attività EMR	Stringa
errorId	ErrorId se l'oggetto non è riuscito.	Stringa
errorMessage	ErrorMessage se l'oggetto non è riuscito.	Stringa
errorStackTrace	Traccia dello stack di errore se l'oggetto non è riuscito.	Stringa
@finishedTime	L'ora in cui è terminata l'esecuzione di questo oggetto.	DateTime
hadoopJobLog	Log delle attività Hadoop disponibili per le attività basate su EMR.	Stringa
@healthStatus	Lo stato di integrità dell'oggetto che riflette l'esito positivo o negativo dell'ultima istanza dell'oggetto che ha raggiunto lo stato di un'istanza terminata.	Stringa
@healthStatusFromInstanceid	Id dell'ultimo oggetto dell'istanza che ha raggiunto lo stato terminato.	Stringa
@healthStatusUpdated Ora	L'ora in cui lo stato di integrità è stato aggiornato o l'ultima volta.	DateTime

Campi Runtime	Description	Tipo di slot
hostname	Il nome host del client che si è aggiudicato il tentativo dell'attività.	Stringa
@lastDeactivatedTime	L'ora in cui l'oggetto è stato disattivato.	DateTime
@latestCompletedRun Ora	L'orario dell'esecuzione più recente durante il quale l'esecuzione è stata completata.	DateTime
@latestRunTime	L'orario dell'esecuzione più recente durante il quale l'esecuzione è stata pianificata.	DateTime
@nextRunTime	L'orario dell'esecuzione da programmare come successiva.	DateTime
reportProgressTime	Il periodo di tempo più recente in cui l'attività remota ha segnalato un progresso.	DateTime
@scheduledEndTime	L'orario di termine della pianificazione per un oggetto	DateTime
@scheduledStartTime	L'orario di inizio della pianificazione per l'oggetto	DateTime
@status	Lo stato di questo oggetto.	Stringa
@version	Versione della pipeline con cui l'oggetto è stato creato.	Stringa
@waitingOn	Descrizione dell'elenco di dipendenze per cui questo oggetto è in attesa.	Oggetto di riferimento, ad esempio «waitingOn»: {"ref":» myRunnableObject Id "}

Campi di sistema	Description	Tipo di slot
@error	Errore che descrive il formato oggetto errato.	Stringa
@pipelineId	L'id della pipeline a cui appartiene questo oggetto.	Stringa
@sphere	La sfera di un oggetto indica la propria posizione nel ciclo di vita: i Component Objects generano Instance Objects che eseguono Attempt Objects.	Stringa

Vedi anche

- [ShellCommandActivity](#)
- [CopyActivity](#)
- [EmrCluster](#)

HiveActivity

Esegue una query Hive su un cluster EMR. `HiveActivity` semplifica la configurazione di un'attività di Amazon EMR e crea automaticamente tabelle Hive in base ai dati di input provenienti da Amazon S3 o Amazon RDS. È sufficiente specificare l'esecuzione di HiveQL sui dati di origine. AWS Data Pipeline crea automaticamente tabelle Hive con `${input1}`, `${input2}` e così via, in base ai campi di input dell'oggetto `HiveActivity`.

Per gli input di Amazon S3, il `dataFormat` campo viene utilizzato per creare i nomi delle colonne Hive.

Per gli input MySQL (Amazon RDS), i nomi delle colonne per la query SQL vengono utilizzati per creare i nomi delle colonne Hive.

Note

Questa attività utilizza il [CSV Serde](#) di Hive.

Esempio

Di seguito è illustrato un esempio di questo tipo di oggetto. Questo oggetto fa riferimento a tre altri oggetti definiti nello stesso file di definizione della pipeline. `MySchedule` è un oggetto `Schedule` e `MyS3Input` e `MyS3Output` sono oggetti di nodi di dati.

```
{
  "name" : "ProcessLogData",
  "id" : "MyHiveActivity",
  "type" : "HiveActivity",
  "schedule" : { "ref": "MySchedule" },
  "hiveScript" : "INSERT OVERWRITE TABLE ${output1} select
host,user,time,request,status,size from ${input1};",
  "input" : { "ref": "MyS3Input" },
  "output" : { "ref": "MyS3Output" },
  "runsOn" : { "ref": "MyEmrCluster" }
}
```

Sintassi

Campi Object Invocation	Description	Tipo di slot
<code>schedule</code>	Questo oggetto viene richiamato entro l'esecuzione di un intervallo di pianificazione. Specificare un riferimento alla pianificazione di un altro oggetto per impostare l'ordine di esecuzione e delle dipendenze per questo oggetto. È possibile soddisfare questo requisito impostando esplicitamente una pianificazione sull'oggetto, ad esempio specificando « <code>schedule</code> »: <code>{"ref": "«"}</code> . <code>DefaultSchedule</code> Nella maggior parte dei casi, è meglio inserire il riferimento alla pianificazione nell'oggetto pipeline di default, in modo che tutti gli oggetti possano ereditare tale pianificazione. O, se la pipeline consiste di una struttura di pianificazioni (nidificate all'interno della pianificazione principale), è possibile creare un oggetto padre che dispone di un	Oggetto di riferimento, ad esempio « <code>schedule</code> »: <code>{"ref": "«myScheduleId"}</code>

Campi Object Invocation	Description	Tipo di slot
	riferimento alla pianificazione. Per ulteriori informazioni sulle configurazioni di pianificazione opzionali di esempio, consulta https://docs.aws.amazon.com/datapipeline/latest/DeveloperGuide/dp-object-schedule.html .	

Gruppo richiesto (uno dei seguenti è obbligatorio)	Description	Tipo di slot
hiveScript	Lo script Hive da eseguire.	Stringa
scriptUri	La posizione dello script Hive da eseguire (ad esempio, s3:// scriptLocation).	Stringa

Gruppo obbligatorio	Description	Tipo di slot
runsOn	Il cluster EMR in cui viene eseguita questa HiveActivity .	Oggetto di riferimento, ad esempio «runSon»: {"ref":» myEmrCluster Id "}
workerGroup	Il gruppo di lavoro. Utilizzato per le attività di routing. Se si fornisce un valore runsOn ed esiste workerGroup , workerGroup verrà ignorato.	Stringa
input	Origine dati di input.	Oggetto di riferimento, ad esempio «input»: {"ref":» myDataNode Id "}

Gruppo obbligatorio	Description	Tipo di slot
output	Origine dati di output.	Oggetto di riferimento, ad esempio «output»: {"ref":» myDataNode Id "}

Campi opzionali	Description	Tipo di slot
attemptStatus	Lo stato segnalato più di recente dall'attività remota.	Stringa
attemptTimeout	Timeout per il completamento del lavoro in remoto. Se questo campo è impostato, un'attività remota che non viene completata entro il tempo impostato di avvio viene tentata di nuovo.	Periodo
dependsOn	Specifica una dipendenza su un altro oggetto eseguibile.	Oggetto di riferimento, ad esempio «dependsOn»: {"ref":» myActivityId «}
failureAndRerunModalità	Descrive il comportamento del nodo consumer quando le dipendenze presentano un errore o vengono di nuovo eseguite.	Enumerazione
hadoopQueue	Il nome della coda del pianificatore Hadoop a cui verrà inviato il processo.	Stringa
lateAfterTimeout	Il tempo trascorso dopo l'inizio della pipeline entro il quale l'oggetto deve essere completato. Viene attivato solo quando il tipo di pianificazione non è impostato su. ondemand	Periodo

Campi opzionali	Description	Tipo di slot
maxActiveInstances	Il numero massimo di istanze attive simultanee di un componente. Le riesecuzioni non contano ai fini del numero di istanze attive.	Numero intero
maximumRetries	Numero massimo di tentativi in caso di errore.	Numero intero
onFail	Un'azione da eseguire quando l'oggetto corrente ha esito negativo.	Oggetto di riferimento, ad esempio «onFail»: {"ref":» myActionId «}
onLateAction	Azioni che devono essere attivate se un oggetto non è stato ancora pianificato o non è ancora completo.	Oggetto di riferimento, ad esempio "onLateAction«: {"ref":» myActionId «}
onSuccess	Un'operazione da eseguire quando l'oggetto corrente ha esito positivo.	Oggetto di riferimento, ad esempio «onSuccess»: {"ref":» myActionId «}
parent	Padre dell'oggetto corrente da cui saranno ereditati gli slot.	Oggetto di riferimento, ad esempio «parent»: {"ref":» myBaseObject Id "}
pipelineLogUri	L'URI S3 (ad esempio 's3://BucketName/Key/ ') per caricare i log per la pipeline.	Stringa
postActivityTaskConfig	Lo script di configurazione post-attività da eseguire. Questo è composto da un URI dello script della shell in Amazon S3 e da un elenco di argomenti.	Oggetto di riferimento, ad esempio "postActivityTaskConfig»: {"ref":» myShellScript ConfigId «}

Campi opzionali	Description	Tipo di slot
preActivityTaskConfig	Lo script di configurazione pre-attività da eseguire. Questo è composto da un URI dello script della shell in Amazon S3 e da un elenco di argomenti.	Oggetto di riferimento, ad esempio "preActivityTaskConfig»: {"ref»:» myShellScript ConfigId «}
precondizione	Definisce eventualmente una precondizione. Un nodo dati non è contrassegnato come "READY" finché tutte le precondizioni non siano state soddisfatte.	Oggetto di riferimento, ad esempio «precondition»: {"ref»:» myPreconditionId «}
reportProgressTimeout	Timeout per chiamate successive di attività in remoto a reportProgress . Se impostato , le attività in remoto che non presentano avanzamenti nel periodo specificato potrebbero essere considerate bloccate e sono quindi oggetto di un altro tentativo.	Periodo
resizeClusterBeforeIn esecuzione	Ridimensiona il cluster prima di eseguire questa attività per adattare i nodi di dati DynamoDB specificati come input o output. <div data-bbox="472 1262 1149 1864" style="border: 1px solid #add8e6; border-radius: 10px; padding: 10px; background-color: #e6f2ff;"> <p> Note</p> <p>Se la tua attività utilizza una <code>DynamoDBDataNode</code> come nodo di dati di input o output e se imposti <code>resizeClusterBeforeRunning</code> su <code>TRUE</code>, AWS Data Pipeline viene avviata utilizzando i tipi di istanza <code>m3.xlarge</code> . Questo sovrascrive le tue scelte in termini di tipi di istanze con <code>m3.xlarge</code> , con un possibile aumento dei costi.</p> </div>	Booleano

Campi opzionali	Description	Tipo di slot
resizeClusterMaxIstanze	Un limite per il numero massimo di istanze che possono essere richieste dall'algoritmo di ridimensionamento.	Numero intero
retryDelay	La durata del timeout tra due tentativi.	Periodo
scheduleType	Il tipo di pianificazione consente di specificare se gli oggetti nella definizione di pipeline devono essere programmati all'inizio o alla fine dell'intervallo. Time Series Style Scheduling significa che le istanze vengono programmate al termine di ogni intervallo e Cron Style Scheduling significa che le istanze vengono programmate all'inizio di ogni intervallo. Una pianificazione on demand consente di eseguire una pipeline una sola volta, per attivazione. Questo significa che non è necessario clonare o ricreare la pipeline per eseguirla di nuovo. Se utilizzi una pianificazione on demand, devi specificarlo nell'oggetto predefinito e deve essere l'unico scheduleType specificato per gli oggetti della pipeline. Per utilizzare le pipeline su richiesta, è sufficiente chiamare l'ActivatePipeline operazione per ogni esecuzione e successiva. I valori sono: cron, ondemand e timeseries.	Enumerazione

Campi opzionali	Description	Tipo di slot
scriptVariable	Specifica le variabili di script per Amazon EMR da passare a Hive durante l'esecuzione di uno script. Ad esempio, le seguenti variabili di script di esempio passano le variabili SAMPLE e FILTER_DATE a Hive: SAMPLE=s3://elasticmapreduce/samples/hive-ads e FILTER_DATE=#{format(@scheduledStartTime, 'YYYY-MM-dd')}% . Questo campo accetta più valori e funziona con entrambi i campi script e scriptUri . Inoltre, scriptVariable funziona indipendentemente dall'impostazione della fase di sviluppo su true o false. Il campo è particolarmente utile per inviare valori dinamici a Hive utilizzando le espressioni e le funzioni AWS Data Pipeline.	Stringa
fase	Stabilisce se è abilitata la gestione temporanea a prima o dopo aver eseguito lo script. Non consentito con Hive 11, quindi usa un'AMI Amazon EMR versione 3.2.0 o successiva.	Booleano

Campi Runtime	Description	Tipo di slot
@activeInstances	Elenco di oggetti di istanze attive attualmente programmate.	Oggetto di riferimento, ad esempio «activeInstances»: {"ref":» Id " } myRunnableObject
@actualEndTime	L'ora in cui è terminata l'esecuzione di questo oggetto.	DateTime

Campi Runtime	Description	Tipo di slot
@actualStartTime	L'ora in cui è stata avviata l'esecuzione di questo oggetto.	DateTime
cancellationReason	CancellationReason se questo oggetto è stato annullato.	Stringa
@cascadeFailedOn	Descrizione della catena di dipendenza che ha generato l'errore dell'oggetto.	Oggetto di riferimento, ad esempio "cascadeFailedOn«: {" ref»:» myRunnableObject Id "
emrStepLog	I log dei passaggi di Amazon EMR sono disponibili solo nei tentativi di attività EMR.	Stringa
errorId	ErrorId se l'oggetto non è riuscito.	Stringa
errorMessage	ErrorMessage se l'oggetto non è riuscito.	Stringa
errorStackTrace	Traccia dello stack di errore se l'oggetto non è riuscito.	Stringa
@finishedTime	L'ora in cui è terminata l'esecuzione di questo oggetto.	DateTime
hadoopJobLog	Log delle attività Hadoop disponibili per le attività basate su EMR.	Stringa
@healthStatus	Lo stato di integrità dell'oggetto che riflette l'esito positivo o negativo dell'ultima istanza dell'oggetto che ha raggiunto lo stato di un'istanza terminata.	Stringa
@healthStatusFromInstanceid	Id dell'ultimo oggetto dell'istanza che ha raggiunto lo stato terminato.	Stringa

Campi Runtime	Description	Tipo di slot
@ Ora healthStatusUpdated	L'ora in cui lo stato di integrità è stato aggiornato o l'ultima volta.	DateTime
hostname	Il nome host del client che si è aggiudicato il tentativo dell'attività.	Stringa
@lastDeactivatedTime	L'ora in cui l'oggetto è stato disattivato.	DateTime
@ latestCompletedRun Ora	L'orario dell'esecuzione più recente durante il quale l'esecuzione è stata completata.	DateTime
@latestRunTime	L'orario dell'esecuzione più recente durante il quale l'esecuzione è stata pianificata.	DateTime
@nextRunTime	L'orario dell'esecuzione da programmare come successiva.	DateTime
reportProgressTime	Il periodo di tempo più recente in cui l'attività remota ha segnalato un progresso.	DateTime
@scheduledEndTime	L'orario di termine della pianificazione per un oggetto.	DateTime
@scheduledStartTime	L'orario di inizio della pianificazione per un oggetto.	DateTime
@status	Lo stato di questo oggetto.	Stringa
@version	Versione della pipeline con cui l'oggetto è stato creato.	Stringa
@waitingOn	Descrizione dell'elenco di dipendenze per cui questo oggetto è in attesa.	Oggetto di riferimento, ad esempio «waitingOn»: {"ref":» myRunnableObject Id "}

Campi di sistema	Description	Tipo di slot
@error	Errore che descrive il formato oggetto errato.	Stringa
@pipelineId	L'id della pipeline a cui appartiene questo oggetto.	Stringa
@sphere	La sfera di un oggetto indica la propria posizione nel ciclo di vita: i Component Objects generano Instance Objects che eseguono Attempt Objects.	Stringa

Vedi anche

- [ShellCommandActivity](#)
- [EmrActivity](#)

HiveCopyActivity

Esegue una query Hive su un cluster EMR. HiveCopyActivity semplifica la copia dei dati tra tabelle DynamoDB. HiveCopyActivity accetta un'istruzione HiveQL per filtrare i dati di input da DynamoDB a livello di colonna e riga.

Esempio

L'esempio seguente mostra come usare HiveCopyActivity e DynamoDBExportDataFormat per copiare i dati da una versione DynamoDBDataNode a un'altra, mentre i dati vengono filtrati, in base a un timestamp.

```
{
  "objects": [
    {
      "id" : "DataFormat.1",
      "name" : "DataFormat.1",
      "type" : "DynamoDBExportDataFormat",
      "column" : "timeStamp BIGINT"
    },
    {
```

```

    "id" : "DataFormat.2",
    "name" : "DataFormat.2",
    "type" : "DynamoDBExportDataFormat"
  },
  {
    "id" : "DynamoDBDataNode.1",
    "name" : "DynamoDBDataNode.1",
    "type" : "DynamoDBDataNode",
    "tableName" : "item_mapped_table_restore_temp",
    "schedule" : { "ref" : "ResourcePeriod" },
    "dataFormat" : { "ref" : "DataFormat.1" }
  },
  {
    "id" : "DynamoDBDataNode.2",
    "name" : "DynamoDBDataNode.2",
    "type" : "DynamoDBDataNode",
    "tableName" : "restore_table",
    "region" : "us_west_1",
    "schedule" : { "ref" : "ResourcePeriod" },
    "dataFormat" : { "ref" : "DataFormat.2" }
  },
  {
    "id" : "EmrCluster.1",
    "name" : "EmrCluster.1",
    "type" : "EmrCluster",
    "schedule" : { "ref" : "ResourcePeriod" },
    "masterInstanceType" : "m1.xlarge",
    "coreInstanceCount" : "4"
  },
  {
    "id" : "HiveTransform.1",
    "name" : "Hive Copy Transform.1",
    "type" : "HiveCopyActivity",
    "input" : { "ref" : "DynamoDBDataNode.1" },
    "output" : { "ref" : "DynamoDBDataNode.2" },
    "schedule" : { "ref" : "ResourcePeriod" },
    "runsOn" : { "ref" : "EmrCluster.1" },
    "filterSql" : "`timeStamp` > unix_timestamp(\"#{@scheduledStartTime}\", \"yyyy-MM-dd'T'HH:mm:ss\")"
  },
  {
    "id" : "ResourcePeriod",
    "name" : "ResourcePeriod",
    "type" : "Schedule",

```

```

    "period" : "1 Hour",
    "startDateTime" : "2013-06-04T00:00:00",
    "endDateTime" : "2013-06-04T01:00:00"
  }
]
}

```

Sintassi

Campi Object Invocation	Description	Tipo di slot
schedule	<p>Questo oggetto viene richiamato entro l'esecuzione di un intervallo di pianificazione. Gli utenti devono specificare un riferimento alla pianificazione di un altro oggetto per impostare l'ordine di esecuzione delle dipendenze per questo oggetto. Gli utenti possono soddisfare questo requisito impostando esplicitamente una pianificazione sull'oggetto, ad esempio specificando «schedule»: {"ref": "«"}. DefaultSchedule Nella maggior parte dei casi, è meglio inserire il riferimento alla pianificazione nell'oggetto pipeline di default, in modo che tutti gli oggetti possano ereditare tale pianificazione. O, se la pipeline consiste di una struttura di pianificazioni (nidificate all'interno della pianificazione principale), gli utenti possono creare un oggetto padre che dispone di un riferimento alla pianificazione. Per ulteriori informazioni sulle configurazioni di pianificazione opzionali di esempio, consulta https://docs.aws.amazon.com/datapipeline/latest/DeveloperGuide/dp-object-schedule.html</p>	<p>Oggetto di riferimento, ad esempio «schedule»: {"ref": «myScheduleId»}</p>

Gruppo richiesto (uno dei seguenti è obbligatorio)	Description	Tipo di slot
runsOn	Specificare il cluster per l'esecuzione.	Oggetto di riferimento, ad esempio «runsOn»: {"ref":» myResourceId «}
workerGroup	Il gruppo di lavoro. Utilizzato per le attività di routing. Se si fornisce un valore runsOn ed esiste workerGroup , workerGroup verrà ignorato.	Stringa

Campi opzionali	Description	Tipo di slot
attemptStatus	Lo stato segnalato più di recente dall'attività remota.	Stringa
attemptTimeout	Il timeout per il completamento del lavoro in remoto. Se questo campo è impostato, un'attività remota che non viene completata entro il tempo impostato di avvio viene tentata di nuovo.	Periodo
dependsOn	Specifica la dipendenza su un altro oggetto eseguibile.	Oggetto di riferimento, ad esempio «dependsOn»: {"ref":» «myActivityId}
failureAndRerunModo	Descrive il comportamento del nodo consumer quando le dipendenze presentano un errore o vengono di nuovo eseguite.	Enumerazione
filterSql	Un frammento di istruzione SQL Hive che filtra un sottoinsieme di dati DynamoDB o	Stringa

Campi opzionali	Description	Tipo di slot
	Amazon S3 da copiare. Il filtro deve contenere solo predicati e non iniziare con una clausola WHERE, perché AWS Data Pipeline la aggiunge automaticamente.	
input	Origine dati di input. Questo deve essere <code>S3DataNode</code> o <code>DynamoDBDataNode</code> . Se utilizzi <code>DynamoDBNode</code> , specifica <code>DynamoDBExportDataFormat</code> .	Oggetto di riferimento, ad esempio «input»: {"ref":» Id "} myDataNode
lateAfterTimeout	Il tempo trascorso dall'inizio della pipeline entro il quale l'oggetto deve essere completato. Viene attivato solo quando il tipo di pianificazione non è impostato su. <code>ondemand</code>	Periodo
maxActiveInstances	Il numero massimo di istanze attive simultanee di un componente. Le riesecuzioni non contano ai fini del numero di istanze attive.	Numero intero
maximumRetries	Numero massimo di tentativi in caso di errore.	Numero intero
onFail	Un'azione da eseguire quando l'oggetto corrente ha esito negativo.	Oggetto di riferimento, ad esempio «onFail»: {"ref":» myActionId «}
onLateAction	Azioni che devono essere attivate se un oggetto non è stato ancora pianificato o non è ancora completo.	Oggetto di riferimento, ad esempio "onLateAction«: {"ref":» myActionId «}
onSuccess	Un'operazione da eseguire quando l'oggetto corrente ha esito positivo.	Oggetto di riferimento, ad esempio «onSuccess»: {"ref":» myActionId «}

Campi opzionali	Description	Tipo di slot
output	Origine dati di output. Se l'input è <code>S3DataNode</code> , questo deve essere <code>DynamoDBDataNode</code> . Altrimenti può essere <code>S3DataNode</code> o <code>DynamoDBDataNode</code> . Se utilizzi <code>DynamoDBNode</code> , specifica <code>DynamoDBExportDataFormat</code> .	Oggetto di riferimento, ad esempio «output»: <code>{"ref":» myDataNodeId "}</code>
parent	Padre dell'oggetto corrente da cui saranno ereditati gli slot.	Oggetto di riferimento, ad esempio «parent»: <code>{"ref":» myBaseObjectId "}</code>
pipelineLogUri	L'URI di Amazon S3, ad esempio, per il <code>'s3://BucketName/Key/'</code> caricamento dei log per la pipeline.	Stringa
postActivityTaskConfig	Lo script di configurazione post-attività da eseguire. Questo è composto da un URI dello script della shell in Amazon S3 e da un elenco di argomenti.	Oggetto di riferimento, ad esempio "postActivityTaskConfig»: <code>{"ref":» myShellScriptConfigId «}</code>
preActivityTaskConfig	Lo script di configurazione pre-attività da eseguire. Questo è composto da un URI dello script della shell in Amazon S3 e da un elenco di argomenti.	Oggetto di riferimento, ad esempio "preActivityTaskConfig»: <code>{"ref":» myShellScriptConfigId «}</code>
precondizione	Definisce eventualmente una precondizione. Un nodo dati non è contrassegnato come "READY" finché tutte le precondizioni non siano state soddisfatte.	Oggetto di riferimento, ad esempio «precondizione»: <code>{"ref":» «myPreconditionId}</code>

Campi opzionali	Description	Tipo di slot
reportProgressTimeout	Timeout per chiamate successive di attività in remoto a <code>reportProgress</code> . Se impostato , le attività in remoto che non presentano avanzamenti nel periodo specificato potrebbero essere considerate bloccate e sono quindi oggetto di un altro tentativo.	Periodo
resizeClusterBeforeInesecuzione	Ridimensiona il cluster prima di eseguire questa attività per adattare i nodi di dati DynamoDB specificati come input o output. <div data-bbox="472 716 1149 1325" style="border: 1px solid #add8e6; border-radius: 10px; padding: 10px; background-color: #e6f2ff;"> <p> Note</p> <p>Se la tua attività utilizza una <code>DynamoDBDataNode</code> come nodo di dati di input o output e se imposti <code>resizeClusterBeforeRunning</code> su <code>TRUE</code>, AWS Data Pipeline viene avviata utilizzando i tipi di istanza <code>m3.xlarge</code> . Questo sovrascrive le tue scelte in termini di tipi di istanze con <code>m3.xlarge</code> , con un possibile aumento dei costi.</p> </div>	Booleano
resizeClusterMaxIstanze	Un limite per il numero massimo di istanze che possono essere richieste dall' algoritmo di ridimensionamento	Numero intero
retryDelay	La durata del timeout tra due tentativi.	Periodo

Campi opzionali	Description	Tipo di slot
scheduleType	<p>Il tipo di pianificazione consente di specificare se gli oggetti nella definizione di pipeline devono essere programmati all'inizio o alla fine dell'intervallo. Time Series Style Scheduling significa che le istanze vengono programmate al termine di ogni intervallo e Cron Style Scheduling significa che le istanze vengono programmate all'inizio di ogni intervallo. Una pianificazione on demand consente di eseguire una pipeline una sola volta, per attivazione. Questo significa che non è necessario clonare o ricreare la pipeline per eseguirla di nuovo. Se utilizzi una pianificazione on demand, devi specificarlo nell'oggetto predefinito e deve essere l'unico scheduleType specificato per gli oggetti della pipeline. Per utilizzare le pipeline su richiesta, è sufficiente chiamare l'ActivatePipeline operazione per ogni esecuzione e successiva. I valori sono: cron, ondemand e timeseries.</p>	Enumerazione
Campi Runtime	Description	Tipo di slot
@activeInstances	Elenco di oggetti di istanze attive attualmente programmate.	Oggetto di riferimento, ad esempio «activeInstances»: {"ref":» Id "}" myRunnableObject
@actualEndTime	L'ora in cui è terminata l'esecuzione di questo oggetto.	DateTime
@actualStartTime	L'ora in cui è stata avviata l'esecuzione di questo oggetto.	DateTime

Campi Runtime	Description	Tipo di slot
cancellationReason	CancellationReason se questo oggetto è stato annullato.	Stringa
@cascadeFailedOn	Descrizione della catena di dipendenza che ha generato l'errore dell'oggetto.	Oggetto di riferimento, ad esempio "cascadeFailedOn«: {" ref»:» myRunnableObject Id "}
emrStepLog	I log dei passaggi di Amazon EMR sono disponibili solo nei tentativi di attività EMR.	Stringa
errorId	ErrorId se l'oggetto non è riuscito.	Stringa
errorMessage	ErrorMessage se l'oggetto non è riuscito.	Stringa
errorStackTrace	Traccia dello stack di errore se l'oggetto non è riuscito.	Stringa
@finishedTime	L'ora in cui è terminata l'esecuzione di questo oggetto.	DateTime
hadoopJobLog	Log delle attività Hadoop disponibili per le attività basate su EMR.	Stringa
@healthStatus	Lo stato di integrità dell'oggetto che riflette l'esito positivo o negativo dell'ultima istanza dell'oggetto che ha raggiunto lo stato di un'istanza terminata.	Stringa
@healthStatusFromInstanceid	Id dell'ultimo oggetto dell'istanza che ha raggiunto lo stato terminato.	Stringa
@ Ora healthStatusUpdated	L'ora in cui lo stato di integrità è stato aggiornato o l'ultima volta.	DateTime

Campi Runtime	Description	Tipo di slot
hostname	Il nome host del client che si è aggiudicato il tentativo dell'attività.	Stringa
@lastDeactivatedTime	L'ora in cui l'oggetto è stato disattivato.	DateTime
@latestCompletedRun Ora	L'orario dell'esecuzione più recente durante il quale l'esecuzione è stata completata.	DateTime
@latestRunTime	L'orario dell'esecuzione più recente durante il quale l'esecuzione è stata pianificata.	DateTime
@nextRunTime	L'orario dell'esecuzione da programmare come successiva.	DateTime
reportProgressTime	Il periodo di tempo più recente in cui l'attività remota ha segnalato un progresso.	DateTime
@scheduledEndTime	L'orario di termine della pianificazione per un oggetto.	DateTime
@scheduledStartTime	L'orario di inizio della pianificazione per l'oggetto.	DateTime
@status	Lo stato di questo oggetto.	Stringa
@version	Versione della pipeline con cui l'oggetto è stato creato.	Stringa
@waitingOn	Descrizione dell'elenco di dipendenze per cui questo oggetto è in attesa.	Oggetto di riferimento, ad esempio «waitingOn»: {"ref":» myRunnableObject Id "}

Campi di sistema	Description	Tipo di slot
@error	Errore che descrive il formato oggetto errato.	Stringa
@pipelineId	L'id della pipeline a cui appartiene questo oggetto.	Stringa
@sphere	La sfera di un oggetto indica la propria posizione nel ciclo di vita: i Component Objects generano Instance Objects che eseguono Attempt Object.	Stringa

Vedi anche

- [ShellCommandActivity](#)
- [EmrActivity](#)

PigActivity

PigActivity fornisce supporto nativo per gli script Pig AWS Data Pipeline senza la necessità di utilizzare `ShellCommandActivity` o `EmrActivity`. Inoltre, PigActivity supporta l'archiviazione dei dati. Quando il campo della fase è impostato su `true`, AWS Data Pipeline gestisce temporaneamente i dati di input come schema in Pig senza codice aggiuntivo da parte dell'utente.

Esempio

La seguente pipeline di esempio mostra come utilizzare PigActivity. La pipeline di esempio esegue le operazioni seguenti:

- MyPigActivity1 carica i dati da Amazon S3 ed esegue uno script Pig che seleziona alcune colonne di dati e li carica su Amazon S3.
- MyPigActivity2 carica il primo output, seleziona alcune colonne e tre righe di dati e lo carica su Amazon S3 come secondo output.
- MyPigActivity3 carica il secondo dato di output, inserisce due righe di dati e solo la colonna denominata «quinta» in Amazon RDS.
- MyPigActivity4 carica i dati Amazon RDS, seleziona la prima riga di dati e la carica su Amazon S3.

```
{
  "objects": [
    {
      "id": "MyInputData1",
      "schedule": {
        "ref": "MyEmrResourcePeriod"
      },
      "directoryPath": "s3://example-bucket/pigTestInput",
      "name": "MyInputData1",
      "dataFormat": {
        "ref": "MyInputDataType1"
      },
      "type": "S3DataNode"
    },
    {
      "id": "MyPigActivity4",
      "scheduleType": "CRON",
      "schedule": {
        "ref": "MyEmrResourcePeriod"
      },
      "input": {
        "ref": "MyOutputData3"
      },
      "pipelineLogUri": "s3://example-bucket/path/",
      "name": "MyPigActivity4",
      "runsOn": {
        "ref": "MyEmrResource"
      },
      "type": "PigActivity",
      "dependsOn": {
        "ref": "MyPigActivity3"
      },
      "output": {
        "ref": "MyOutputData4"
      },
      "script": "B = LIMIT ${input1} 1; ${output1} = FOREACH B GENERATE one;",
      "stage": "true"
    },
    {
      "id": "MyPigActivity3",
      "scheduleType": "CRON",
      "schedule": {
        "ref": "MyEmrResourcePeriod"
      }
    }
  ]
}
```

```
    },
    "input": {
      "ref": "MyOutputData2"
    },
    "pipelineLogUri": "s3://example-bucket/path",
    "name": "MyPigActivity3",
    "runsOn": {
      "ref": "MyEmrResource"
    },
    "script": "B = LIMIT ${input1} 2; ${output1} = FOREACH B GENERATE Fifth;",
    "type": "PigActivity",
    "dependsOn": {
      "ref": "MyPigActivity2"
    },
    "output": {
      "ref": "MyOutputData3"
    },
    "stage": "true"
  },
  {
    "id": "MyOutputData2",
    "schedule": {
      "ref": "MyEmrResourcePeriod"
    },
    "name": "MyOutputData2",
    "directoryPath": "s3://example-bucket/PigActivityOutput2",
    "dataFormat": {
      "ref": "MyOutputDataType2"
    },
    "type": "S3DataNode"
  },
  {
    "id": "MyOutputData1",
    "schedule": {
      "ref": "MyEmrResourcePeriod"
    },
    "name": "MyOutputData1",
    "directoryPath": "s3://example-bucket/PigActivityOutput1",
    "dataFormat": {
      "ref": "MyOutputDataType1"
    },
    "type": "S3DataNode"
  },
  {
```

```

    "id": "MyInputDataType1",
    "name": "MyInputDataType1",
    "column": [
      "First STRING",
      "Second STRING",
      "Third STRING",
      "Fourth STRING",
      "Fifth STRING",
      "Sixth STRING",
      "Seventh STRING",
      "Eighth STRING",
      "Ninth STRING",
      "Tenth STRING"
    ],
    "inputRegex": "^(\\\\S+) (\\\\S+) (\\\\S+) (\\\\S+) (\\\\S+) (\\\\S+) (\\\\S+) (\\\\S+) (\\\\S+) (\\\\S+)",
    "type": "Regex"
  },
  {
    "id": "MyEmrResource",
    "region": "us-east-1",
    "schedule": {
      "ref": "MyEmrResourcePeriod"
    },
    "keyPair": "example-keypair",
    "masterInstanceType": "m1.small",
    "enableDebugging": "true",
    "name": "MyEmrResource",
    "actionOnTaskFailure": "continue",
    "type": "EmrCluster"
  },
  {
    "id": "MyOutputDataType4",
    "name": "MyOutputDataType4",
    "column": "one STRING",
    "type": "CSV"
  },
  {
    "id": "MyOutputData4",
    "schedule": {
      "ref": "MyEmrResourcePeriod"
    },
    "directoryPath": "s3://example-bucket/PigActivityOutput3",
    "name": "MyOutputData4",

```

```

    "dataFormat": {
      "ref": "MyOutputDataType4"
    },
    "type": "S3DataNode"
  },
  {
    "id": "MyOutputDataType1",
    "name": "MyOutputDataType1",
    "column": [
      "First STRING",
      "Second STRING",
      "Third STRING",
      "Fourth STRING",
      "Fifth STRING",
      "Sixth STRING",
      "Seventh STRING",
      "Eighth STRING"
    ],
    "columnSeparator": "*",
    "type": "Custom"
  },
  {
    "id": "MyOutputData3",
    "username": "__",
    "schedule": {
      "ref": "MyEmrResourcePeriod"
    },
    "insertQuery": "insert into #{table} (one) values (?)",
    "name": "MyOutputData3",
    "*password": "__",
    "runsOn": {
      "ref": "MyEmrResource"
    },
    "connectionString": "jdbc:mysql://example-database-instance:3306/example-database",
    "selectQuery": "select * from #{table}",
    "table": "example-table-name",
    "type": "MySQLDataNode"
  },
  {
    "id": "MyOutputDataType2",
    "name": "MyOutputDataType2",
    "column": [
      "Third STRING",

```

```

    "Fourth STRING",
    "Fifth STRING",
    "Sixth STRING",
    "Seventh STRING",
    "Eighth STRING"
  ],
  "type": "TSV"
},
{
  "id": "MyPigActivity2",
  "scheduleType": "CRON",
  "schedule": {
    "ref": "MyEmrResourcePeriod"
  },
  "input": {
    "ref": "MyOutputData1"
  },
  "pipelineLogUri": "s3://example-bucket/path",
  "name": "MyPigActivity2",
  "runsOn": {
    "ref": "MyEmrResource"
  },
  "dependsOn": {
    "ref": "MyPigActivity1"
  },
  "type": "PigActivity",
  "script": "B = LIMIT ${input1} 3; ${output1} = FOREACH B GENERATE Third, Fourth,
Fifth, Sixth, Seventh, Eighth;",
  "output": {
    "ref": "MyOutputData2"
  },
  "stage": "true"
},
{
  "id": "MyEmrResourcePeriod",
  "startDateTime": "2013-05-20T00:00:00",
  "name": "MyEmrResourcePeriod",
  "period": "1 day",
  "type": "Schedule",
  "endDateTime": "2013-05-21T00:00:00"
},
{
  "id": "MyPigActivity1",
  "scheduleType": "CRON",

```

```

    "schedule": {
      "ref": "MyEmrResourcePeriod"
    },
    "input": {
      "ref": "MyInputData1"
    },
    "pipelineLogUri": "s3://example-bucket/path",
    "scriptUri": "s3://example-bucket/script/pigTestScript.q",
    "name": "MyPigActivity1",
    "runsOn": {
      "ref": "MyEmrResource"
    },
    "scriptVariable": [
      "column1=First",
      "column2=Second",
      "three=3"
    ],
    "type": "PigActivity",
    "output": {
      "ref": "MyOutputData1"
    },
    "stage": "true"
  }
]
}

```

Il contenuto di `pigTestScript.q` è il seguente.

```

B = LIMIT ${input1} $three; ${output1} = FOREACH B GENERATE $column1, $column2, Third,
Fourth, Fifth, Sixth, Seventh, Eighth;

```

Sintassi

Campi Object Invocation	Description	Tipo di slot
schedule	Questo oggetto viene richiamato entro l'esecuzione di un intervallo di pianificazione. Gli utenti devono specificare un riferimento alla pianificazione di un altro oggetto per impostare l'ordine di esecuzione delle dipendenze per	Oggetto di riferimento, ad esempio, «schedule»: {"ref":» myScheduleId «}

Campi Object Invocation	Description	Tipo di slot
	<p>questo oggetto. Gli utenti possono soddisfare e questo requisito impostando esplicitamente una pianificazione sull'oggetto, ad esempio specificando «schedule»: {"ref": "«}. DefaultSchedule Nella maggior parte dei casi, è meglio inserire il riferimento alla pianificazione nell'oggetto pipeline di default, in modo che tutti gli oggetti possano ereditare tale pianificazione. O, se la pipeline consiste di una struttura di pianificazioni (nidificate all'interno della pianificazione principale), gli utenti possono creare un oggetto padre che dispone di un riferimento alla pianificazione. Per ulteriori informazioni sulle configurazioni di pianificazione opzionali di esempio, consulta https://docs.aws.amazon.com/datapipeline/latest/DeveloperGuide/dp-object-schedule.html</p>	
Gruppo richiesto (uno dei seguenti è obbligatorio)	Description	Tipo di slot
script	Lo script Pig da eseguire	Stringa
scriptUri	La posizione dello script Pig da eseguire (ad esempio, s3:// scriptLocation).	Stringa

Gruppo richiesto (uno dei seguenti è obbligatorio)	Description	Tipo di slot
runsOn	Cluster EMR su cui viene eseguito. PigActivity	Oggetto di riferimento, ad esempio «runSon»: {"ref":» myEmrCluster Id "}
workerGroup	Il gruppo di lavoro. Utilizzato per le attività di routing. Se si fornisce un valore runsOn ed esiste workerGroup , workerGroup verrà ignorato.	Stringa

Campi opzionali	Description	Tipo di slot
attemptStatus	Lo stato segnalato più di recente dall'attività remota.	Stringa
attemptTimeout	Il timeout per il completamento del lavoro in remoto. Se questo campo è impostato, un'attività remota che non viene completata entro il tempo impostato di avvio viene tentata di nuovo.	Periodo
dependsOn	Specifica la dipendenza su un altro oggetto eseguibile.	Oggetto di riferimento, ad esempio «dependsOn»: {"ref":» myActivityId «}
failureAndRerunModo	Descrive il comportamento del nodo consumer quando le dipendenze presentano un errore o vengono di nuovo eseguite.	Enumerazione

Campi opzionali	Description	Tipo di slot
input	Origine dati di input.	Oggetto di riferimento, ad esempio, «input»: {"ref":» myDataNode Id "}
lateAfterTimeout	Il tempo trascorso dall'inizio della pipeline entro il quale l'oggetto deve essere completato. Viene attivato solo quando il tipo di pianificazione non è impostato su. ondemand	Periodo
maxActiveInstances	Il numero massimo di istanze attive simultanee di un componente. Le riesecuzioni non contano ai fini del numero di istanze attive.	Numero intero
maximumRetries	Numero massimo di tentativi in caso di errore.	Numero intero
onFail	Un'azione da eseguire quando l'oggetto corrente ha esito negativo.	Oggetto di riferimento, ad esempio «onFail»: {"ref":» myActionId «}
onLateAction	Azioni che devono essere attivate se un oggetto non è stato ancora pianificato o non è ancora completo.	Oggetto di riferimento, ad esempio "onLateAction«: {"ref":» myActionId «}
onSuccess	Un'operazione da eseguire quando l'oggetto corrente ha esito positivo.	Oggetto di riferimento, ad esempio «onSuccess»: {"ref":» myActionId «}
output	Origine dati di output.	Oggetto di riferimento, ad esempio «output»: {"ref":» myDataNode Id "}

Campi opzionali	Description	Tipo di slot
parent	Padre dell'oggetto corrente da cui saranno ereditati gli slot.	Oggetto di riferimento, ad esempio «parent»: {"ref":» myBaseObject Id "}
pipelineLogUri	L'URI di Amazon S3 (ad esempio 's3://BucketName/Key/ ') per caricare i log per la pipeline.	Stringa
postActivityTaskConfig	Lo script di configurazione post-attività da eseguire. È costituito da un URI dello script di shell in Amazon S3 e da un elenco di argomenti.	Oggetto di riferimento, ad esempio "postActivityTaskConfig»: {"ref":» myShellScript ConfigId «}
preActivityTaskConfig	Lo script di configurazione pre-attività da eseguire. Questo è composto da un URI dello script della shell in Amazon S3 e da un elenco di argomenti.	Oggetto di riferimento, ad esempio "preActivityTaskConfig»: {"ref":» myShellScript ConfigId «}
precondizione	Definisce eventualmente una precondizione. Un nodo dati non è contrassegnato come "READY" finché tutte le precondizioni non siano state soddisfatte.	Oggetto di riferimento, ad esempio, «precondition»: {"ref":» myPreconditionId «}
reportProgressTimeout	Timeout per chiamate successive di attività in remoto a reportProgress . Se impostato , le attività in remoto che non presentano avanzamenti nel periodo specificato potrebbero essere considerate bloccate e sono quindi oggetto di un altro tentativo.	Periodo

Campi opzionali	Description	Tipo di slot
<code>resizeClusterBeforeInEsecuzione</code>	<p>Ridimensiona il cluster prima di eseguire questa attività per adattare i nodi di dati DynamoDB specificati come input o output.</p> <div data-bbox="472 401 1151 1003" style="border: 1px solid #add8e6; border-radius: 10px; padding: 10px;"><p> Note</p><p>Se la tua attività utilizza una <code>DynamoDBDataNode</code> come nodo di dati di input o output e se imposti <code>resizeClusterBeforeRunning</code> su <code>TRUE</code>, AWS Data Pipeline viene avviata utilizzando i tipi di istanza <code>m3.xlarge</code>. Questo sovrascrive le tue scelte in termini di tipi di istanze con <code>m3.xlarge</code>, con un possibile aumento dei costi.</p></div>	Booleano
<code>resizeClusterMaxIstanze</code>	Un limite per il numero massimo di istanze che possono essere richieste dall'algoritmo di ridimensionamento.	Numero intero
<code>retryDelay</code>	La durata del timeout tra due tentativi.	Periodo

Campi opzionali	Description	Tipo di slot
scheduleType	Il tipo di pianificazione consente di specificare se gli oggetti nella definizione di pipeline devono essere programmati all'inizio o alla fine dell'intervallo. Time Series Style Scheduling significa che le istanze vengono programmate al termine di ogni intervallo e Cron Style Scheduling significa che le istanze vengono programmate all'inizio di ogni intervallo. Una pianificazione on demand consente di eseguire una pipeline una sola volta, per attivazione. Questo significa che non è necessario clonare o ricreare la pipeline per eseguirla di nuovo. Se utilizzi una pianificazione on demand, devi specificarlo nell'oggetto predefinito e deve essere l'unico scheduleType specificato per gli oggetti della pipeline. Per utilizzare le pipeline su richiesta, è sufficiente chiamare l'ActivatePipeline operazione per ogni esecuzione e successiva. I valori sono: cron, ondemand e timeseries.	Enumerazione
scriptVariable	Gli argomenti da passare allo script Pig. È possibile utilizzare scriptVariable con lo script o scriptUri.	Stringa
fase	Stabilisce se è abilitata la gestione temporanea e consente allo script Pig di accedere alle tabelle dei dati gestiti temporaneamente, ad esempio \${INPUT1} e \${OUTPUT1}.	Booleano

Campi Runtime	Description	Tipo di slot
@activeInstances	Elenco di oggetti di istanze attive attualmente programmate.	Oggetto di riferimento, ad esempio «activeIn

Campi Runtime	Description	Tipo di slot
		stances»: {"ref»:» Id "} myRunnableObject
@actualEndTime	L'ora in cui è terminata l'esecuzione di questo oggetto.	DateTime
@actualStartTime	L'ora in cui è stata avviata l'esecuzione di questo oggetto.	DateTime
cancellationReason	CancellationReason se questo oggetto è stato annullato.	Stringa
@cascadeFailedOn	Descrizione della catena di dipendenza che ha generato l'errore dell'oggetto.	Oggetto di riferimento, ad esempio "cascadeFailedOn«: {" ref»:» myRunnableObject Id "}
emrStepLog	I log dei passaggi di Amazon EMR sono disponibili solo nei tentativi di attività EMR.	Stringa
errorId	ErrorId se l'oggetto non è riuscito.	Stringa
errorMessage	ErrorMessage se l'oggetto non è riuscito.	Stringa
errorStackTrace	Traccia dello stack di errore se l'oggetto non è riuscito.	Stringa
@finishedTime	L'ora in cui è terminata l'esecuzione di questo oggetto.	DateTime
hadoopJobLog	Log delle attività Hadoop disponibili per le attività basate su EMR.	Stringa
@healthStatus	Lo stato di integrità dell'oggetto che riflette l'esito positivo o negativo dell'ultima istanza dell'oggetto che ha raggiunto lo stato di un'istanza terminata.	Stringa

Campi Runtime	Description	Tipo di slot
@healthStatusFromInstanceid	Id dell'ultimo oggetto dell'istanza che ha raggiunto lo stato terminato.	Stringa
@ Ora healthStatusUpdated	L'ora in cui lo stato di integrità è stato aggiornato o l'ultima volta.	DateTime
hostname	Il nome host del client che si è aggiudicato il tentativo dell'attività.	Stringa
@lastDeactivatedTime	L'ora in cui l'oggetto è stato disattivato.	DateTime
@ latestCompletedRun Ora	L'orario dell'esecuzione più recente durante il quale l'esecuzione è stata completata.	DateTime
@latestRunTime	L'orario dell'esecuzione più recente durante il quale l'esecuzione è stata pianificata.	DateTime
@nextRunTime	L'orario dell'esecuzione da programmare come successiva.	DateTime
reportProgressTime	Il periodo di tempo più recente in cui l'attività remota ha segnalato un progresso.	DateTime
@scheduledEndTime	L'orario di termine della pianificazione per l'oggetto.	DateTime
@scheduledStartTime	L'orario di inizio della pianificazione per l'oggetto.	DateTime
@status	Lo stato di questo oggetto.	Stringa
@version	Versione della pipeline con cui l'oggetto è stato creato.	Stringa

Campi Runtime	Description	Tipo di slot
@waitingOn	Descrizione dell'elenco di dipendenze per cui questo oggetto è in attesa.	Oggetto di riferimento, ad esempio «waitingOn»: {"ref":» myRunnableObject Id "}
Campi di sistema	Description	Tipo di slot
@error	Errore che descrive il formato oggetto errato.	Stringa
@pipelineId	L'ID della pipeline a cui appartiene questo oggetto.	Stringa
@sphere	La sfera di un oggetto indica la propria posizione nel ciclo di vita: i Component Objects generano Instance Objects che eseguono Attempt Objects.	Stringa

Vedi anche

- [ShellCommandActivity](#)
- [EmrActivity](#)

RedshiftCopyActivity

Copia i dati da DynamoDB o Amazon S3 su Amazon Redshift. È possibile caricare i dati in una nuova tabella, oppure unirli facilmente alla tabella esistente.

Questa è una panoramica di un caso d'uso in cui utilizzare RedshiftCopyActivity:

1. Inizia a AWS Data Pipeline utilizzarlo per lo staging dei dati in Amazon S3.
2. RedshiftCopyActivityUtilizzalo per spostare i dati da Amazon RDS e Amazon EMR ad Amazon Redshift.

In questo modo puoi caricare i tuoi dati in Amazon Redshift dove puoi analizzarli.

3. [SqlActivity](#) Utilizzalo per eseguire query SQL sui dati che hai caricato in Amazon Redshift.

Inoltre, `RedshiftCopyActivity` consente di lavorare con un `S3DataNode`, poiché supporta un file manifest. Per ulteriori informazioni, consulta [S3 DataNode](#).

Esempio

Di seguito è illustrato un esempio di questo tipo di oggetto.

Per garantire la conversione dei formati, questo esempio utilizza i parametri speciali di conversione [EMPTYASNULL](#) e [IGNOREBLANKLINES](#) in `commandOptions`. Per informazioni, consulta i [parametri di conversione dei dati](#) nella Amazon Redshift Database Developer Guide.

```
{
  "id" : "S3ToRedshiftCopyActivity",
  "type" : "RedshiftCopyActivity",
  "input" : { "ref": "MyS3DataNode" },
  "output" : { "ref": "MyRedshiftDataNode" },
  "insertMode" : "KEEP_EXISTING",
  "schedule" : { "ref": "Hour" },
  "runsOn" : { "ref": "MyEc2Resource" },
  "commandOptions": ["EMPTYASNULL", "IGNOREBLANKLINES"]
}
```

L'esempio seguente di definizione di pipeline mostra un'attività che utilizza la modalità di inserimento APPEND:

```
{
  "objects": [
    {
      "id": "CSVId1",
      "name": "DefaultCSV1",
      "type": "CSV"
    },
    {
      "id": "RedshiftDatabaseId1",
      "databaseName": "dbname",
      "username": "user",
      "name": "DefaultRedshiftDatabase1",
      "*password": "password",
    }
  ]
}
```

```

    "type": "RedshiftDatabase",
    "clusterId": "redshiftclusterId"
  },
  {
    "id": "Default",
    "scheduleType": "timeseries",
    "failureAndRerunMode": "CASCADE",
    "name": "Default",
    "role": "DataPipelineDefaultRole",
    "resourceRole": "DataPipelineDefaultResourceRole"
  },
  {
    "id": "RedshiftDataNodeId1",
    "schedule": {
      "ref": "ScheduleId1"
    },
    "tableName": "orders",
    "name": "DefaultRedshiftDataNode1",
    "createTableSql": "create table StructuredLogs (requestBeginTime CHAR(30)
PRIMARY KEY DISTKEY SORTKEY, requestEndTime CHAR(30), hostname CHAR(100), requestDate
varchar(20));",
    "type": "RedshiftDataNode",
    "database": {
      "ref": "RedshiftDatabaseId1"
    }
  },
  {
    "id": "Ec2ResourceId1",
    "schedule": {
      "ref": "ScheduleId1"
    },
    "securityGroups": "MySecurityGroup",
    "name": "DefaultEc2Resource1",
    "role": "DataPipelineDefaultRole",
    "logUri": "s3://myLogs",
    "resourceRole": "DataPipelineDefaultResourceRole",
    "type": "Ec2Resource"
  },
  {
    "id": "ScheduleId1",
    "startDateTime": "yyyy-mm-ddT00:00:00",
    "name": "DefaultSchedule1",
    "type": "Schedule",
    "period": "period",

```

```

    "endTime": "yyyy-mm-ddT00:00:00"
  },
  {
    "id": "S3DataNodeId1",
    "schedule": {
      "ref": "ScheduleId1"
    },
    "filePath": "s3://datapipeline-us-east-1/samples/hive-ads-samples.csv",
    "name": "DefaultS3DataNode1",
    "dataFormat": {
      "ref": "CSVId1"
    },
    "type": "S3DataNode"
  },
  {
    "id": "RedshiftCopyActivityId1",
    "input": {
      "ref": "S3DataNodeId1"
    },
    "schedule": {
      "ref": "ScheduleId1"
    },
    "insertMode": "APPEND",
    "name": "DefaultRedshiftCopyActivity1",
    "runsOn": {
      "ref": "Ec2ResourceId1"
    },
    "type": "RedshiftCopyActivity",
    "output": {
      "ref": "RedshiftDataNodeId1"
    }
  }
]
}

```

L'operazione APPEND aggiunge gli elementi a una tabella indipendentemente dalle chiavi di ordinamento o primarie. Ad esempio, se si ha la seguente tabella, è possibile aggiungere un record con lo stesso ID e valore utente.

ID(PK)	USER
1	aaa
2	bbb

È possibile aggiungere un record con lo stesso ID e valore utente:

ID(PK)	USER
1	aaa
2	bbb
1	aaa

Note

Se un'operazione APPEND viene interrotta e riprovata, la risultante pipeline rieseguita viene potenzialmente aggiunta dall'inizio. L'operazione potrebbe causare un ulteriore doppione, quindi è necessario essere a conoscenza di questo comportamento, soprattutto se si ha una logica che conteggia il numero di righe.

Per un tutorial, vedere [Copia i dati su Amazon Redshift utilizzando AWS Data Pipeline.](#)

Sintassi

Campi obbligatori	Description	Tipo di slot
insertMode	<p>Stabilisce che cosa AWS Data Pipeline fa con i dati pre-esistenti nella tabella di destinazione che si sovrappone alle righe dei dati da caricare.</p> <p>I valori validi sono: KEEP_EXISTING , OVERWRITE_EXISTING , TRUNCATE e APPEND.</p> <p>KEEP_EXISTING aggiunge nuove righe alla tabella, mentre non modifica le righe esistenti.</p> <p>KEEP_EXISTING e OVERWRITE_EXISTING utilizzano la chiave primaria, l'ordinamento e le chiavi di distribuzione per identificare quali righe in entrata abbinare alle righe esistenti. Consulta Aggiornamento e</p>	Enumerazione

Campi obbligatori	Description	Tipo di slot
	<p>inserimento di nuovi dati nella Amazon Redshift Database Developer Guide.</p> <p>TRUNCATE elimina tutti i dati della tabella di destinazione prima di scrivere i nuovi dati.</p> <p>APPEND aggiunge tutti i record alla fine della tabella Redshift. APPEND non richiede una chiave primaria e di distribuzione o la chiave di ordinamento, in modo da poter aggiungere gli elementi che potrebbero essere potenziali duplicati.</p>	

Campi Object Invocation	Description	Tipo di slot
schedule	<p>Questo oggetto viene richiamato entro l'esecuzione di un intervallo di pianificazione.</p> <p>Specificare un riferimento alla pianificazione di un altro oggetto per impostare l'ordine di esecuzione delle dipendenze per questo oggetto.</p> <p>Nella maggior parte dei casi, è preferibile inserire il riferimento alla pianificazione nell'oggetto pipeline di default, in modo che tutti gli oggetti possano ereditare tale pianificazione. Ad esempio, è possibile impostare una pianificazione esplicitamente sull'oggetto, specificando <code>"schedule": {"ref": "DefaultSchedule"}</code>.</p> <p>Se la pianificazione master nella pipeline contiene pianificazioni nidificate, è possibile</p>	<p>Reference Object, ad esempio:</p> <pre>"schedule": {"ref": "myScheduleId"}</pre>

Campi Object Invocation	Description	Tipo di slot
	<p>creare un oggetto padre che dispone di un riferimento alla pianificazione.</p> <p>Per ulteriori informazioni sulle configurazioni di pianificazione opzionali di esempio, consulta Pianificazione.</p>	

Gruppo richiesto (uno dei seguenti è obbligatorio)	Description	Tipo di slot
runsOn	Le risorse di calcolo per eseguire l'attività o il comando. Ad esempio, un'istanza Amazon EC2 o un cluster Amazon EMR.	Oggetto di riferimento, ad esempio «runSon»: {"ref":» myResourceId «}
workerGroup	Il gruppo di lavoro. Utilizzato per le attività di routing. Se si fornisce un valore runsOn ed esiste workerGroup , workerGroup verrà ignorato.	Stringa

Campi opzionali	Description	Tipo di slot
attemptStatus	Lo stato segnalato più di recente dall'attività remota.	Stringa
attemptTimeout	Timeout per il completamento del lavoro in remoto. Se questo campo è impostato, un'attività remota che non viene completata entro il tempo impostato di avvio viene tentata di nuovo.	Periodo

Campi opzionali	Description	Tipo di slot
commandOptions	<p>Richiede i parametri da passare al nodo di dati Amazon Redshift durante l'COPYoperazione. Per informazioni sui parametri, consulta COPY nella Amazon Redshift Database Developer Guide.</p> <p>Mentre carica la tabella, COPY tenta di convertire e in modo implicito le stringhe al tipo di dati della colonna di destinazione. Oltre alle conversioni predefinite dei dati che si verificano automaticamente, se si ricevono errori o altre esigenze di conversione, è possibile specificare i parametri di conversione aggiuntivi. Per informazioni, consulta i parametri di conversione dei dati nella Amazon Redshift Database Developer Guide.</p> <p>Se un formato di dati è associato al nodo di dati in ingresso o in uscita, allora i parametri forniti vengono ignorati.</p> <p>Poiché l'operazione di copia utilizza per prima cosa COPY per inserire i dati in una tabella intermedia, quindi usa un comando INSERT per copiare i dati dalla tabella intermedia nella tabella di destinazione, alcuni parametri COPY non sono applicabili, ad esempio la capacità del comando COPY di abilitare la compressione automatica della tabella. Se la compressione è obbligatoria, aggiungi i dettagli di codifica della colonna all'istruzione CREATE TABLE.</p> <p>Inoltre, in alcuni casi, quando deve scaricare dati dal cluster Amazon Redshift e creare file in Amazon S3, si affida RedshiftCopyActivi</p>	Stringa

Campi opzionali	Description	Tipo di slot
	<p>ty al funzionamento UNLOAD di Amazon Redshift.</p> <p>Per migliorare le prestazioni durante la copia e lo scaricamento, specificare il parametro PARALLEL OFF del comando UNLOAD. Per informazioni sui parametri, consulta UNLOAD nella Amazon Redshift Database Developer Guide.</p>	
dependsOn	Specifica una dipendenza su un altro oggetto eseguibile.	Reference Object: "dependsOn": { "ref": "myActivityId" }
failureAndRerunModalità	Descrive il comportamento del nodo consumer quando le dipendenze presentano un errore o vengono di nuovo eseguite	Enumerazione
input	Nodo dei dati di input. L'origine dati può essere Amazon S3, DynamoDB o Amazon Redshift.	Reference Object: "input": { "ref": "myDataNodeId" }
lateAfterTimeout	Il tempo trascorso dopo l'inizio della pipeline entro il quale l'oggetto deve essere completato. Viene attivato solo quando il tipo di pianificazione non è impostato su. ondemand	Periodo
maxActiveInstances	Il numero massimo di istanze attive simultanee di un componente. Le riesecuzioni non contano ai fini del numero di istanze attive.	Numero intero
maximumRetries	Numero massimo di tentativi in caso di errore	Numero intero

Campi opzionali	Description	Tipo di slot
onFail	Un'azione da eseguire quando l'oggetto corrente ha esito negativo.	Reference Object: "onFail": { "ref": "myActionId" }
onLateAction	Azioni che devono essere attivate se un oggetto non è stato ancora pianificato o non è ancora completo.	Reference Object: "onLateAction": { "ref": "myActionId" }
onSuccess	Un'operazione da eseguire quando l'oggetto corrente ha esito positivo.	Reference Object: "onSuccess": { "ref": "myActionId" }
output	Nodo dei dati di output. Il percorso di output può essere Amazon S3 o Amazon Redshift.	Reference Object: "output": { "ref": "myDataNodeId" }
parent	Padre dell'oggetto corrente da cui saranno ereditati gli slot.	Reference Object: "parent": { "ref": "myBaseObjectId" }
pipelineLogUri	L'URI S3 (ad esempio 's3://BucketName/Key/ ') per caricare i log per la pipeline.	Stringa
precondizione	Definisce eventualmente una precondizione. Un nodo dati non è contrassegnato come "READY" finché tutte le precondizioni non siano state soddisfatte.	Reference Object: "precondition": { "ref": "myPreconditionId" }

Campi opzionali	Description	Tipo di slot
<code>coda</code>	<p>Corrisponde all'<code>query_group</code> impostazione di Amazon Redshift, che consente di assegnare e dare priorità alle attività simultanee in base alla loro collocazione nelle code.</p> <p>Amazon Redshift limita il numero di connessioni simultanee a 15. Per ulteriori informazioni, consulta Assigning Queries to Queues nella Amazon RDS Database Developer Guide.</p>	Stringa
<code>reportProgressTimeout</code>	<p>Timeout per chiamate successive di attività in remoto a <code>reportProgress</code>.</p> <p>Se impostato, le attività in remoto che non presentano avanzamenti nel periodo specificato potrebbero essere considerate bloccate e sono quindi oggetto di un altro tentativo.</p>	Periodo
<code>retryDelay</code>	La durata del timeout tra due tentativi.	Periodo

Campi opzionali	Description	Tipo di slot
<code>scheduleType</code>	<p>Consente di specificare se la pianificazione per gli oggetti è nella pipeline. I valori sono <code>cron</code>, <code>ondemand</code> e <code>timeseries</code> .</p> <p>La pianificazione <code>timeseries</code> significa che le istanze sono programmate al termine di ogni intervallo.</p> <p>La pianificazione <code>Cron</code> significa che le istanze sono programmate all'inizio di ogni intervallo.</p> <p>Una pianificazione <code>ondemand</code> consente di eseguire una pipeline una sola volta, per attivazione. Questo significa che non è necessario clonare o ricreare la pipeline per eseguirla di nuovo.</p> <p>Per utilizzare le pipeline <code>ondemand</code>, chiama l'operazione <code>ActivatePipeline</code> per ogni esecuzione successiva.</p> <p>Se utilizzi una pianificazione <code>ondemand</code>, devi specificarlo nell'oggetto predefinito e deve essere l'unica <code>scheduleType</code> specificata per gli oggetti della pipeline.</p>	Enumerazione

Campi opzionali	Description	Tipo di slot
<code>transformSql</code>	<p>L'espressione SQL <code>SELECT</code> utilizzata per trasformare i dati di input.</p> <p>Esegui l'espressione <code>transformSql</code> nella tabella denominata <code>staging</code>.</p> <p>Quando copi dati da DynamoDB o Amazon S3AWS Data Pipeline, crea una tabella chiamata «staging» e inizialmente carica i dati al suo interno. I dati di questa tabella vengono utilizzati per aggiornare la tabella di destinazione.</p> <p>Lo schema di output di <code>transformSql</code> deve corrispondere allo schema della tabella di destinazione finale.</p> <p>Se si specifica l'opzione <code>transformSql</code>, viene creata una seconda tabella intermedia dall'istruzione SQL specificata. I dati di questa seconda tabella intermedia vengono quindi aggiornati nella tabella di destinazione finale.</p>	Stringa

Campi Runtime	Description	Tipo di slot
<code>@activeInstances</code>	Elenco di oggetti di istanze attive attualmente programmate.	Reference Object: "activeInstances": { "ref": "myRunnable ObjectId" }
<code>@actualEndTime</code>	L'ora in cui è terminata l'esecuzione di questo oggetto.	DateTime

Campi Runtime	Description	Tipo di slot
@actualStartTime	L'ora in cui è stata avviata l'esecuzione di questo oggetto.	DateTime
cancellationReason	CancellationReason se questo oggetto è stato annullato.	Stringa
@cascadeFailedOn	Descrizione della catena di dipendenza che ha generato l'errore dell'oggetto.	Reference Object: "cascadeFailedOn": { "ref": "myRunnable ObjectId" }
emrStepLog	Log della fase EMR disponibili solo sui tentativi delle attività EMR	Stringa
errorId	ErrorId se l'oggetto non è riuscito.	Stringa
errorMessage	ErrorMessage se l'oggetto non è riuscito.	Stringa
errorStackTrace	Traccia dello stack di errore se l'oggetto non è riuscito.	Stringa
@finishedTime	L'ora in cui è terminata l'esecuzione di questo oggetto.	DateTime
hadoopJobLog	Log delle attività Hadoop disponibili per le attività basate su EMR.	Stringa
@healthStatus	Lo stato di integrità dell'oggetto che riflette l'esito positivo o negativo dell'ultima istanza dell'oggetto che ha raggiunto lo stato di un'istanza terminata.	Stringa
@healthStatusFromInstanceid	Id dell'ultimo oggetto dell'istanza che ha raggiunto lo stato terminato.	Stringa

Campi Runtime	Description	Tipo di slot
@ Ora healthStatusUpdated	L'ora in cui lo stato di integrità è stato aggiornato o l'ultima volta.	DateTime
hostname	Il nome host del client che si è aggiudicato il tentativo dell'attività.	Stringa
@lastDeactivatedTime	L'ora in cui l'oggetto è stato disattivato.	DateTime
@ latestCompletedRun Ora	L'orario dell'esecuzione più recente durante il quale l'esecuzione è stata completata.	DateTime
@latestRunTime	L'orario dell'esecuzione più recente durante il quale l'esecuzione è stata pianificata.	DateTime
@nextRunTime	L'orario dell'esecuzione da programmare come successiva.	DateTime
reportProgressTime	Il periodo di tempo più recente in cui l'attività remota ha segnalato un progresso.	DateTime
@scheduledEndTime	L'orario di termine della pianificazione per un oggetto.	DateTime
@scheduledStartTime	L'orario di inizio della pianificazione per l'oggetto.	DateTime
@status	Lo stato di questo oggetto.	Stringa
@version	Versione della pipeline con cui l'oggetto è stato creato.	Stringa
@waitingOn	Descrizione dell'elenco di dipendenze per cui questo oggetto è in attesa.	Reference Object: "waitingOn": { "ref": "myRunnableObjectId" }

Campi di sistema	Description	Tipo di slot
@error	Errore che descrive il formato oggetto errato.	Stringa
@pipelineId	L'id della pipeline a cui appartiene questo oggetto.	Stringa
@sphere	La sfera di un oggetto. Indica la propria posizione nel ciclo di vita. Ad esempio, i Component Objects generano Instance Objects che eseguono Attempt Objects.	Stringa

ShellCommandActivity

Consente di eseguire un comando o uno script. È possibile usare `ShellCommandActivity` per eseguire operazioni pianificate di tipo Cron o con serie temporali.

Quando il `stage` campo è impostato su `true` e utilizzato con un `S3DataNode`, `ShellCommandActivity` supporta il concetto di staging dei dati, il che significa che puoi spostare i dati da Amazon S3 a una posizione di stage, ad esempio Amazon EC2 o il tuo ambiente locale, eseguire operazioni sui dati utilizzando script e `ShellCommandActivity` poi spostarli nuovamente su Amazon S3.

In questo caso, quando il comando shell è connesso a un input `S3DataNode`, gli script shell operano direttamente sui dati utilizzando `${INPUT1_STAGING_DIR}`, `${INPUT2_STAGING_DIR}` e altri campi, con riferimento ai campi di input `ShellCommandActivity`.

Allo stesso modo, l'output del comando shell può essere archiviato in una directory di output per essere inviato automaticamente ad Amazon S3, a cui si fa riferimento da `${OUTPUT1_STAGING_DIR}` e così via. `${OUTPUT2_STAGING_DIR}`

Queste espressioni possono passare come argomenti della riga di comando al comando shell per l'utilizzo in logiche di trasformazione dei dati.

`ShellCommandActivity` restituisce codici e stringhe di errore in stile Linux. Se una `ShellCommandActivity` presenta un errore, l'`error` restituito è un valore diverso da zero.

Esempio

Di seguito è illustrato un esempio di questo tipo di oggetto.

```
{
  "id" : "CreateDirectory",
  "type" : "ShellCommandActivity",
  "command" : "mkdir new-directory"
}
```

Sintassi

Campi Object Invocation	Description	Tipo di slot
schedule	<p>Questo oggetto viene richiamato entro l'esecuzione di un intervallo di <code>schedule</code>.</p> <p>Per impostare l'ordine di esecuzione delle dipendenze per questo oggetto, specificare un riferimento <code>schedule</code> a un altro oggetto.</p> <p>Per soddisfare questo requisito, impostare esplicitamente un <code>schedule</code> sull'oggetto, ad esempio, specificando <code>"schedule": {"ref": "DefaultSchedule"}</code>.</p> <p>Nella maggior parte dei casi, è preferibile inserire il riferimento <code>schedule</code> all'oggetto pipeline di default, in modo che tutti gli oggetti ereditano tale pianificazione. Se la pipeline consiste di una struttura di pianificazioni (nidificate all'interno della pianificazione principale), è possibile creare un oggetto padre che dispone di un riferimento alla pianificazione.</p>	<p>Oggetto di riferimento, ad esempio</p> <pre>«schedule»: {"ref":» «} myScheduleId</pre>

Campi Object Invocation	Description	Tipo di slot
	<p>Per distribuire il carico, AWS Data Pipeline crea oggetti fisici leggermente prima del previsto, ma li eseguite secondo la pianificazione.</p> <p>Per ulteriori informazioni sulle configurazioni di pianificazione opzionali di esempio, consulta https://docs.aws.amazon.com/datapipeline/latest/DeveloperGuide/dp-object-schedule.html</p>	
Gruppo richiesto (uno dei seguenti è obbligatorio)	Description	Tipo di slot
command	Il comando da eseguire. Utilizzare \$ per fare riferimento ai parametri posizionali e <code>scriptArgument</code> per specificare i parametri del comando. Questo valore ed eventuali parametri associati devono funzionare nell'ambiente da cui si sta eseguendo il Task Runner.	Stringa
scriptUri	Un percorso URI di Amazon S3 per un file da scaricare ed eseguire come comando shell. Specifica solo uno <code>scriptUri</code> o solo un campo <code>command</code> . Se <code>scriptUri</code> non è in grado di utilizzare i parametri, utilizzare <code>command</code> .	Stringa

Gruppo richiesto (uno dei seguenti è obbligatorio)	Description	Tipo di slot
runsOn	La risorsa di calcolo per eseguire l'attività o il comando, ad esempio un'istanza Amazon EC2 o un cluster Amazon EMR.	Oggetto di riferimento, ad esempio «runOn»: {"ref":» myResourceId «}
workerGroup	Utilizzato per le attività di routing. Se si fornisce un valore runsOn ed esiste workerGroup , workerGroup verrà ignorato.	Stringa

Campi opzionali	Description	Tipo di slot
attemptStatus	Lo stato segnalato più di recente dall'attività remota.	Stringa
attemptTimeout	Il timeout per il completamento del lavoro in remoto. Se questo campo è impostato, un'attività remota che non viene completata entro il tempo impostato di avvio viene tentata di nuovo.	Periodo
dependsOn	Specifica una dipendenza su un altro oggetto eseguibile.	Oggetto di riferimento, ad esempio «dependsOn»: {"ref":» «myActivityId}
failureAndRerunModo	Descrive il comportamento del nodo consumer quando le dipendenze presentano un errore o vengono di nuovo eseguite.	Enumerazione
input	Posizione dei dati di input.	Oggetto di riferimento, ad esempio «input»:

Campi opzionali	Description	Tipo di slot
		<code>{"ref":» myDataNode Id "}</code>
<code>lateAfterTimeout</code>	Il tempo trascorso dall'inizio della pipeline entro il quale l'oggetto deve essere completato. Viene attivato solo quando il tipo di pianificazione non è impostato su. <code>ondemand</code>	Periodo
<code>maxActiveInstances</code>	Il numero massimo di istanze attive simultanee di un componente. Le riesecuzioni non contano ai fini del numero di istanze attive.	Numero intero
<code>maximumRetries</code>	Numero massimo di tentativi in caso di errore.	Numero intero
<code>onFail</code>	Un'azione da eseguire quando l'oggetto corrente ha esito negativo.	Oggetto di riferimento, ad esempio <code>«onFail»: {"ref":» myActionId «}</code>
<code>onLateAction</code>	Azioni che devono essere attivate se un oggetto non è stato ancora pianificato o non è completo.	Oggetto di riferimento, ad esempio <code>"onLateAction«: {"ref":» myActionId «}</code>
<code>onSuccess</code>	Un'operazione da eseguire quando l'oggetto corrente ha esito positivo.	Oggetto di riferimento, ad esempio <code>«onSuccess»: {"ref":» myActionId «}</code>
<code>output</code>	Posizione dei dati di output.	Oggetto di riferimento, ad esempio <code>«output»: {"ref":» myDataNode Id "}</code>
<code>parent</code>	Padre dell'oggetto corrente da cui saranno ereditati gli slot.	Oggetto di riferimento, ad esempio <code>«parent»: {"ref":» myBaseObj ect Id "}</code>

Campi opzionali	Description	Tipo di slot
pipelineLogUri	L'URI di Amazon S3, ad esempio 's3://BucketName/Key/' per il caricamento dei log per la pipeline.	Stringa
precondizione	Definisce eventualmente una precondizione. Un nodo dati non è contrassegnato come "READY" finché tutte le precondizioni non siano state soddisfatte.	Oggetto di riferimento, ad esempio «precondition»: {"ref":» «} myPreconditionId
reportProgressTimeout	Il timeout per chiamate successive a <code>reportProgress</code> da parte di attività in remoto. Se impostato, le attività in remoto che non presentano avanzamenti nel periodo specificato potrebbero essere considerate bloccate e sono quindi oggetto di un altro tentativo.	Periodo
retryDelay	La durata del timeout tra due tentativi.	Periodo

Campi opzionali	Description	Tipo di slot
<code>scheduleType</code>	<p>Consente di specificare se gli oggetti nella definizione di pipeline devono essere programmati all'inizio o alla fine dell'intervallo.</p> <p>I valori possibili sono: <code>cron</code>, <code>ondemand</code> e <code>timeseries</code> .</p> <p>Se le istanze sono impostate su <code>timeseries</code> significa che sono programmate al termine di ogni intervallo.</p> <p>Se le istanze sono impostate su <code>Cron</code> significa che sono programmate all'inizio di ogni intervallo.</p> <p>Se sono impostate su <code>ondemand</code>, è possibile eseguire una pipeline una sola volta, per attivazione. Questo significa che non è necessario clonare o ricreare la pipeline per eseguirla di nuovo. Se utilizzi una pianificazione <code>ondemand</code>, devi specificarlo nell'oggetto predefinito come l'unico <code>scheduleType</code> per gli oggetti della pipeline. Per utilizzare le pipeline <code>ondemand</code>, chiama l'operazione <code>ActivatePipeline</code> per ogni esecuzione successiva.</p>	Enumerazione

Campi opzionali	Description	Tipo di slot
scriptArgument	<p>Un array di stringhe in formato JSON da passare al comando specificato dal comando. Ad esempio, se il comando è <code>echo \$1 \$2</code>, specificare <code>scriptArgument</code> come <code>"param1", "param2"</code>. Per più argomenti e parametri, passare <code>scriptArgument</code> come segue: <code>"scriptArgument": "arg1", "scriptArgument": "param1", "scriptArgument": "arg2", "scriptArgument": "param2"</code>.</p> <p><code>scriptArgument</code> può essere utilizzato solo con <code>command</code>; se si utilizza con <code>scriptUri</code> viene generato un errore.</p>	Stringa
fase	<p>Stabilisce se è abilitata la gestione temporanea e consente ai comandi shell di accedere alle variabili dei dati gestiti temporaneamente, ad esempio <code>\${INPUT1_STAGING_DIR}</code> e <code>\${OUTPUT1_STAGING_DIR}</code>.</p>	Booleano
stderr	<p>Il percorso che riceve messaggi di errore del sistema reindirizzati dal comando. Se utilizzi il <code>runsOn</code> campo, deve trattarsi di un percorso Amazon S3 a causa della natura transitoria della risorsa che esegue la tua attività. Tuttavia, se specifichi il campo <code>workerGroup</code>, viene autorizzato un percorso file locale.</p>	Stringa
stdout	<p>Il percorso Amazon S3 che riceve l'output reindirizzato dal comando. Se utilizzi il <code>runsOn</code> campo, deve trattarsi di un percorso Amazon S3 a causa della natura transitoria della risorsa che esegue la tua attività. Tuttavia, se specifichi il campo <code>workerGroup</code>, viene autorizzato un percorso file locale.</p>	Stringa

Campi Runtime	Description	Tipo di slot
@activeInstances	L'elenco di oggetti di istanze attive attualmente programmate.	Oggetto di riferimento, ad esempio «activeInstances»: {"ref":» Id "}" myRunnableObject
@actualEndTime	L'ora in cui è terminata l'esecuzione di questo oggetto.	DateTime
@actualStartTime	L'ora in cui è stata avviata l'esecuzione di questo oggetto.	DateTime
cancellationReason	cancellationReason se questo oggetto è stato annullato.	Stringa
@cascadeFailedOn	La descrizione della catena di dipendenza che ha generato l'errore dell'oggetto.	Oggetto di riferimento, ad esempio "cascadeFailedOn«: {" ref":» myRunnableObject Id "}"
emrStepLog	I log dei passaggi di Amazon EMR sono disponibili solo per i tentativi di attività di Amazon EMR.	Stringa
errorId	errorId se l'oggetto non riuscito.	Stringa
errorMessage	errorMessage se l'oggetto non riuscito.	Stringa
errorStackTrace	Traccia dello stack di errore se l'oggetto non è riuscito.	Stringa
@finishedTime	L'ora in cui è terminata l'esecuzione dell'oggetto.	DateTime
hadoopJobLog	Registri di lavoro Hadoop disponibili sui tentativi di attività basate su Amazon EMR.	Stringa

Campi Runtime	Description	Tipo di slot
@healthStatus	Lo stato di integrità dell'oggetto che riflette l'esito positivo o negativo dell'ultima istanza dell'oggetto che ha raggiunto lo stato di un'istanza terminata.	Stringa
@healthStatusFromInstanceid	L'Id dell'ultimo oggetto dell'istanza che ha raggiunto lo stato terminato.	Stringa
@ Ora healthStatusUpdated	L'ora in cui lo stato di integrità è stato aggiornato o l'ultima volta.	DateTime
hostname	Il nome host del client che si è aggiudicato il tentativo dell'attività.	Stringa
@lastDeactivatedTime	L'ora in cui l'oggetto è stato disattivato.	DateTime
@ latestCompletedRun Ora	L'orario dell'esecuzione più recente durante il quale l'esecuzione è stata completata.	DateTime
@latestRunTime	L'orario dell'esecuzione più recente durante il quale l'esecuzione è stata pianificata.	DateTime
@nextRunTime	L'orario dell'esecuzione da programmare come successiva.	DateTime
reportProgressTime	Il periodo di tempo più recente in cui l'attività remota ha segnalato un progresso.	DateTime
@scheduledEndTime	L'orario di termine della pianificazione per l'oggetto.	DateTime
@scheduledStartTime	L'orario di inizio della pianificazione per l'oggetto.	DateTime
@status	Lo stato dell'oggetto.	Stringa

Campi Runtime	Description	Tipo di slot
@version	La versione di AWS Data Pipeline utilizzata per creare l'oggetto.	Stringa
@waitingOn	La descrizione dell'elenco di dipendenze per cui questo oggetto è in attesa.	Oggetto di riferimento, ad esempio «waitingOn»: {"ref":» myRunnableObject Id "}

Campi di sistema	Description	Tipo di slot
@error	Errore che descrive l'oggetto con il formato errato.	Stringa
@pipelineId	L'id della pipeline a cui appartiene questo oggetto.	Stringa
@sphere	La posizione di un oggetto nel ciclo di vita. I Component Objects generano Instance Objects che eseguono Attempt Objects.	Stringa

Vedi anche

- [CopyActivity](#)
- [EmrActivity](#)

SqlActivity

Esegue una query SQL (script) su un database.

Esempio

Di seguito è illustrato un esempio di questo tipo di oggetto.

```
{
  "id" : "MySQLActivity",
  "type" : "SqlActivity",
  "database" : { "ref": "MyDatabaseID" },
  "script" : "SQLQuery" | "scriptUri" : s3://scriptBucket/query.sql,
  "schedule" : { "ref": "MyScheduleID" },
}
```

Sintassi

Campi obbligatori	Description	Tipo di slot
database	Il database su cui eseguire lo script SQL fornito.	Oggetto di riferimento, ad esempio «database»: {"ref":» myDatabaseId «}

Campi Object Invocation	Description	Tipo di slot
schedule	<p>Questo oggetto viene richiamato entro l'esecuzione di un intervallo di pianificazione. È necessario specificare un riferimento alla pianificazione di un altro oggetto per impostare l'ordine di esecuzione delle dipendenze per questo oggetto. È possibile impostare una pianificazione esplicitamente sull'oggetto, ad esempio, specificando "schedule": {"ref": "DefaultSchedule"} .</p> <p>Nella maggior parte dei casi, è meglio inserire il riferimento alla pianificazione nell'oggetto pipeline di default, in modo che tutti gli oggetti possano ereditare tale pianificazione.</p>	Oggetto di riferimento, ad esempio «schedule»: {"ref":» myScheduleId «}

Campi Object Invocation	Description	Tipo di slot
	Se la pipeline dispone di una struttura di pianificazioni nidificate all'interno della pianificazione principale, è possibile creare un oggetto padre che dispone di un riferimento alla pianificazione. Per ulteriori informazioni sulle configurazioni di pianificazione opzionali di esempio, consulta https://docs.aws.amazon.com/datapipeline/latest/DeveloperGuide/dp-object-schedule.html	

Gruppo richiesto (uno dei seguenti è obbligatorio)	Description	Tipo di slot
script	Lo script SQL da eseguire. È necessario specificare lo script o lo scriptUri. Quando lo script viene archiviato in Amazon S3, lo script non viene valutato come espressione. Specificare più valori per ScriptArgument è utile quando lo script è archiviato in Amazon S3.	Stringa
scriptUri	Un URI che specifica il percorso di uno script SQL da eseguire in questa attività.	Stringa

Gruppo richiesto (uno dei seguenti è obbligatorio)	Description	Tipo di slot
runsOn	Le risorse di calcolo per eseguire l'attività o il comando. Ad esempio, un'istanza Amazon EC2 o un cluster Amazon EMR.	Oggetto di riferimento, ad esempio

Gruppo richiesto (uno dei seguenti è obbligatorio)	Description	Tipo di slot
		«runSon»: {"ref":» myResourceId «}
workerGroup	Il gruppo di lavoro. Utilizzato per le attività di routing. Se si fornisce un valore runSon ed esiste workerGroup , workerGroup verrà ignorato.	Stringa

Campi opzionali	Description	Tipo di slot
attemptStatus	Lo stato segnalato più di recente dall'attività remota.	Stringa
attemptTimeout	Timeout per il completamento del lavoro in remoto. Se questo campo è impostato, un'attività remota che non viene completata entro il tempo impostato di avvio viene tentata di nuovo.	Periodo
dependsOn	Specifica una dipendenza su un altro oggetto eseguibile.	Oggetto di riferimento, ad esempio «dependsOn»: {"ref":» «myActivityId}
failureAndRerunModo	Descrive il comportamento del nodo consumer quando le dipendenze presentano un errore o vengono di nuovo eseguite	Enumerazione
input	Posizione dei dati di input.	Oggetto di riferimento, ad esempio «input»: {"ref":» myDataNodeId "}

Campi opzionali	Description	Tipo di slot
lateAfterTimeout	Il periodo di tempo dall'inizio programmato della pipeline all'interno del quale deve essere avviata l'esecuzione dell'oggetto.	Periodo
maxActiveInstances	Il numero massimo di istanze attive simultanee di un componente. Le riesecuzioni non contano ai fini del numero di istanze attive.	Numero intero
maximumRetries	Numero massimo di tentativi in caso di errore	Numero intero
onFail	Un'azione da eseguire quando l'oggetto corrente ha esito negativo.	Oggetto di riferimento, ad esempio «onFail»: {"ref":» myActionId «}
onLateAction	Azioni che devono essere attivate se un oggetto non è ancora stato pianificato o non è ancora stato completato nel periodo di tempo trascorso dall'inizio programmato della pipeline, come specificato da ". lateAfterTimeout	Oggetto di riferimento, ad esempio "onLateAction«: {"ref":» myActionId «}
onSuccess	Un'operazione da eseguire quando l'oggetto corrente ha esito positivo.	Oggetto di riferimento, ad esempio «onSuccess»: {"ref":» myActionId «}
output	Posizione dei dati di output. Questo è utile solo per fare riferimento dall'interno di uno script (ad esempio#{output.tablename}) e per creare la tabella di output impostando 'createTableSql' nel nodo dati di output. L'output della query SQL non è scritto nel nodo dei dati di output.	Oggetto di riferimento, ad esempio «output»: {"ref":» myDataNodeId "}

Campi opzionali	Description	Tipo di slot
parent	Padre dell'oggetto corrente da cui saranno ereditati gli slot.	Oggetto di riferimento, ad esempio «parent»: {"ref":» myBaseObject Id "}
pipelineLogUri	L'URI S3 (come 's3://BucketName/Key/ ') per caricare i log per la pipeline.	Stringa
precondizione	Definisce eventualmente una precondizione. Un nodo dati non è contrassegnato come "READY" finché tutte le precondizioni non siano state soddisfatte.	Oggetto di riferimento, ad esempio «precondizione»: {"ref":» «} myPreconditionId
coda	[solo Amazon Redshift] Corrisponde all'impostazione query_group in Amazon Redshift che consente di assegnare e stabilire le priorità di attività simultanee in base al loro posizionamento nelle code. Amazon Redshift limita il numero di connessioni simultanee a 15. Per ulteriori informazioni, consulta Assegnazione di query alle code nella Guida per gli sviluppatori di database Amazon Redshift.	Stringa
reportProgressTimeout	Timeout per chiamate successive di attività in remoto a reportProgress. Se impostato, le attività in remoto che non presentano avanzamenti nel periodo specificato potrebbero essere considerate bloccate e sono quindi oggetto di un altro tentativo.	Periodo
retryDelay	La durata del timeout tra due tentativi.	Periodo

Campi opzionali	Description	Tipo di slot
scheduleType	<p>Il tipo di pianificazione consente di specificare se gli oggetti nella definizione di pipeline devono essere programmati all'inizio o alla fine dell'intervallo. I valori sono <code>cron</code>, <code>ondemand</code> e <code>timeseries</code> .</p> <p>La pianificazione <code>timeseries</code> significa che le istanze sono programmate al termine di ogni intervallo.</p> <p>La pianificazione <code>cron</code> significa che le istanze sono programmate all'inizio di ogni intervallo.</p> <p>Una pianificazione <code>ondemand</code> consente di eseguire una pipeline una sola volta, per attivazione. Questo significa che non è necessario clonare o ricreare la pipeline per eseguirla di nuovo. Se utilizzi una pianificazione <code>ondemand</code>, devi specificarlo nell'oggetto predefinito e deve essere l'unico <code>scheduleType</code> specificato per gli oggetti della pipeline. Per utilizzare le pipeline <code>ondemand</code>, chiama l'operazione <code>ActivatePipeline</code> per ogni esecuzione successiva.</p>	Enumerazione
scriptArgument	<p>Un elenco di variabili per lo script. In alternativa, è possibile inserire le espressioni direttamente nel campo dello script. Per <code>scriptArgument</code> sono utili più valori quando lo script viene memorizzato in Amazon S3. Esempio: <code># {format (@scheduledStartTime, «YY-MM-DD HH:MM:SS")\n# {format (PlusPeriod (@, «1 giorno»), «YY-MM-DD HH:MM:SS")} scheduledStartTime</code></p>	Stringa

Campi Runtime	Description	Tipo di slot
@activeInstances	Elenco di oggetti di istanze attive attualmente programmate.	Oggetto di riferimento, ad esempio «myRunnableObject activeInstances»: {"ref":» Id "}
@actualEndTime	L'ora in cui è terminata l'esecuzione di questo oggetto.	DateTime
@actualStartTime	L'ora in cui è stata avviata l'esecuzione di questo oggetto.	DateTime
cancellationReason	CancellationReason se questo oggetto è stato annullato.	Stringa
@cascadeFailedOn	Descrizione della catena di dipendenza che ha generato l'errore dell'oggetto.	Oggetto di riferimento, ad esempio "cascadeFailedOn«: {" ref»:» myRunnableObject Id "}
emrStepLog	Log della fase EMR disponibili solo sui tentativi delle attività EMR	Stringa
errorId	ErrorId se l'oggetto non è riuscito.	Stringa
errorMessage	ErrorMessage se l'oggetto non è riuscito.	Stringa
errorStackTrace	Traccia dello stack di errore se l'oggetto non è riuscito.	Stringa
@finishedTime	L'ora in cui è terminata l'esecuzione di questo oggetto.	DateTime
hadoopJobLog	Log delle attività Hadoop disponibili per le attività basate su EMR.	Stringa

Campi Runtime	Description	Tipo di slot
@healthStatus	Lo stato di integrità dell'oggetto che riflette l'esito positivo o negativo dell'ultima istanza dell'oggetto che ha raggiunto lo stato di un'istanza terminata.	Stringa
@healthStatusFromInstanceid	Id dell'ultimo oggetto dell'istanza che ha raggiunto lo stato terminato.	Stringa
@healthStatusUpdated Ora	L'ora in cui lo stato di integrità è stato aggiornato o l'ultima volta.	DateTime
hostname	Il nome host del client che si è aggiudicato il tentativo dell'attività.	Stringa
@lastDeactivatedTime	L'ora in cui l'oggetto è stato disattivato.	DateTime
@latestCompletedRun Ora	L'orario dell'esecuzione più recente durante il quale l'esecuzione è stata completata.	DateTime
@latestRunTime	L'orario dell'esecuzione più recente durante il quale l'esecuzione è stata pianificata.	DateTime
@nextRunTime	L'orario dell'esecuzione da programmare come successiva.	DateTime
reportProgressTime	Il periodo di tempo più recente in cui l'attività remota ha segnalato un progresso.	DateTime
@scheduledEndTime	L'orario di termine della pianificazione per un oggetto	DateTime
@scheduledStartTime	L'orario di inizio della pianificazione per l'oggetto	DateTime
@status	Lo stato di questo oggetto.	Stringa

Campi Runtime	Description	Tipo di slot
@version	Versione della pipeline con cui l'oggetto è stato creato.	Stringa
@waitingOn	Descrizione dell'elenco di dipendenze per cui questo oggetto è in attesa.	Oggetto di riferimento, ad esempio «waitingOn»: {"ref":» myRunnableObject Id "}

Campi di sistema	Description	Tipo di slot
@error	Errore che descrive il formato oggetto errato.	Stringa
@pipelineId	L'id della pipeline a cui appartiene questo oggetto.	Stringa
@sphere	La sfera di un oggetto indica la propria posizione nel ciclo di vita: i Component Objects generano Instance Objects che eseguono Attempt Objects.	Stringa

Risorse

Di seguito sono elencati gli oggetti risorse di AWS Data Pipeline:

Oggetti

- [Ec2Resource](#)
- [EmrCluster](#)
- [HttpProxy](#)

Ec2Resource

Un'istanza Amazon EC2 che esegue il lavoro definito da un'attività di pipeline.

AWS Data Pipeline ora supporta IMDSv2 per l'istanza Amazon EC2, che utilizza un metodo orientato alla sessione per gestire meglio l'autenticazione durante il recupero delle informazioni sui metadati dalle istanze. Una sessione inizia e termina una serie di richieste che il software in esecuzione su un'istanza Amazon EC2 utilizza per accedere ai metadati e alle credenziali dell'istanza Amazon EC2 archiviati localmente. Il software avvia una sessione con una semplice richiesta HTTP PUT a IMDSv2. IMDSv2 restituisce un token segreto al software in esecuzione sull'istanza Amazon EC2, che utilizzerà il token come password per effettuare richieste a IMDSv2 di metadati e credenziali.

Note

Per utilizzare IMDSv2 per la tua istanza Amazon EC2, dovrai modificare le impostazioni, poiché l'AMI predefinita non è compatibile con IMDSv2. Puoi specificare una nuova versione AMI che puoi recuperare tramite il seguente parametro SSM: `/aws/service/ami-amazon-linux-latest/amzn-ami-hvm-x86_64-ebs`

Per informazioni sulle istanze Amazon EC2 predefinite che vengono AWS Data Pipeline create se non specifichi un'istanza, consulta [Istanze Amazon EC2 predefinite per regione AWS](#)

Esempi

EC2-Classic

Important

Solo AWS gli account creati prima del 4 dicembre 2013 supportano la piattaforma EC2-Classic. Se disponi di uno di questi account, potresti avere la possibilità di creare oggetti EC2Resource per una pipeline in una rete EC2-Classic anziché in un VPC. Ti consigliamo vivamente di creare risorse per tutte le tue pipeline in VPC. Inoltre, se disponi di risorse esistenti in EC2-Classic, ti consigliamo di migrarle su un VPC.

L'oggetto di esempio seguente avvia un'istanza EC2 in EC2-Classic, con alcuni campi opzionali impostati.

```
{
  "id" : "MyEC2Resource",
  "type" : "Ec2Resource",
```

```

"actionOnTaskFailure" : "terminate",
"actionOnResourceFailure" : "retryAll",
"maximumRetries" : "1",
"instanceType" : "m5.large",
"securityGroups" : [
  "test-group",
  "default"
],
"keyPair" : "my-key-pair"
}

```

EC2-VPC

L'oggetto di esempio seguente avvia un'istanza EC2 in un VPC non di default, con alcuni campi opzionali impostati.

```

{
  "id" : "MyEC2Resource",
  "type" : "Ec2Resource",
  "actionOnTaskFailure" : "terminate",
  "actionOnResourceFailure" : "retryAll",
  "maximumRetries" : "1",
  "instanceType" : "m5.large",
  "securityGroupIds" : [
    "sg-12345678",
    "sg-12345678"
  ],
  "subnetId": "subnet-12345678",
  "associatePublicIpAddress": "true",
  "keyPair" : "my-key-pair"
}

```

Sintassi

Campi obbligatori	Description	Tipo di slot
resourceRole	Il ruolo IAM che controlla le risorse a cui può accedere l'istanza Amazon EC2.	Stringa
role	Il ruolo IAM AWS Data Pipeline utilizzato per creare l'istanza EC2.	Stringa

Campi Object Invocation	Description	Tipo di slot
schedule	<p>Questo oggetto viene richiamato entro l'esecuzione di un intervallo di pianificazione.</p> <p>Per impostare l'ordine di esecuzione delle dipendenze per questo oggetto, specificare un riferimento di pianificazione a un altro oggetto. Questa operazione può essere eseguita in uno dei seguenti modi:</p> <ul style="list-style-type: none">• Per garantire che tutti gli oggetti nella pipeline possano ereditare la pianificazione, impostare una pianificazione sull'oggetto esplicitamente: <code>"schedule": {"ref": "DefaultSchedule"}</code> . Nella maggior parte dei casi, è utile inserire il riferimento alla pianificazione nell'oggetto pipeline di default, in modo che tutti gli oggetti possano ereditare tale pianificazione.• Se la pipeline dispone di pianificazioni nidificate all'interno della pianificazione principale, è possibile creare un oggetto padre che dispone di un riferimento alla pianificazione. Per ulteriori informazioni sulle configurazioni di pianificazione opzionali di esempio, consulta https://docs.aws.amazon.com/datapipeline/latest/DeveloperGuide/dp-object-schedule.html	Oggetto di riferimento, ad esempio <code>"schedule": {"ref": "myScheduleId"}</code>

Campi opzionali	Description	Tipo di slot
actionOnResourceFailure	L'operazione intrapresa dopo il fallimento di una risorsa per questa risorsa. I valori validi sono "retryall" e "retrynone" .	Stringa
actionOnTaskFailure	L'operazione intrapresa dopo il fallimento di un'attività per questa risorsa. I valori validi sono "continue" e "terminate" .	Stringa
associatePublicIpAddress	Indica se assegnare automaticamente un indirizzo IP pubblico all'istanza. Se l'istanza è in Amazon EC2 o Amazon VPC, il valore predefinito è true. In caso contrario, il valore predefinito è false.	Booleano
attemptStatus	Lo stato segnalato più di recente dall'attività remota.	Stringa
attemptTimeout	Timeout per il completamento del lavoro in remoto. Se questo campo è impostato, un'attività remota che non viene completata entro il tempo impostato di avvio viene tentata di nuovo.	Periodo
availabilityZone	La zona di disponibilità in cui avviare l'istanza Amazon EC2.	Stringa
Disattiva IMDS V1	Il valore predefinito è false e abilita sia IMDSv1 che IMDSv2. Se lo imposti su true, disabilita IMDSv1 e fornisce solo IMDSv2	Booleano
failureAndRerunModality	Descrive il comportamento del nodo consumer quando le dipendenze presentano un errore o vengono di nuovo eseguite.	Enumerazione
httpProxy	Host proxy che i clienti utilizzano per connettersi ai servizi AWS.	Oggetto di riferimento, ad esempio "httpProxy":

Campi opzionali	Description	Tipo di slot
		<code>{"ref": "myHttpProxyId"}</code>
imageId	L'ID dell'AMI utilizzato per l'istanza. Per impostazione predefinita, AWS Data Pipeline utilizza il tipo di virtualizzazione AMI HVM. Gli ID AMI specifici utilizzati sono basati su una regione. È possibile sovrascrivere l'AMI predefinita specificando l'AMI HVM di tua scelta. Per ulteriori informazioni sui tipi di AMI, consulta la sezione Tipi di virtualizzazione delle AMI Linux e Ricerca di un'AMI Linux nella Guida per l'utente di Amazon EC2 per le istanze Linux.	Stringa
initTimeout	Il tempo di attesa prima dell'avvio della risorsa.	Periodo
instanceCount	Obsoleta.	Numero intero
instanceType	Il tipo di istanza Amazon EC2 da avviare.	Stringa
keyPair	Nome della coppia di chiavi. Se avvii un'istanza a Amazon EC2 senza specificare una coppia di key pair, non puoi accedervi.	Stringa
lateAfterTimeout	Il tempo trascorso dall'inizio della pipeline entro il quale l'oggetto deve essere completato. Viene attivato solo quando il tipo di pianificazione non è impostato su. <code>ondemand</code>	Periodo
maxActiveInstances	Il numero massimo di istanze attive simultanee di un componente. Le riesecuzioni non contano ai fini del numero di istanze attive.	Numero intero
maximumRetries	Numero massimo di tentativi in caso di errore.	Numero intero
minInstanceCount	Obsoleta.	Numero intero

Campi opzionali	Description	Tipo di slot
onFail	Un'azione da eseguire quando l'oggetto corrente ha esito negativo.	Oggetto di riferimento, ad esempio "onFail": { "ref": "myActionId" }
onLateAction	Azioni che devono essere attivate se un oggetto non è stato ancora pianificato o se è ancora in esecuzione.	Oggetto di riferimento, ad esempio "onLateAction": { "ref": "myActionId" }
onSuccess	Un'operazione da eseguire quando l'oggetto corrente ha esito positivo.	Oggetto di riferimento, ad esempio "onSuccess": { "ref": "myActionId" }
parent	Padre dell'oggetto corrente da cui vengono ereditati gli slot.	Oggetto di riferimento, ad esempio "parent": { "ref": "myBaseObjectId" }
pipelineLogUri	L'URI di Amazon S3 (ad esempio 's3://BucketName/Key/') per il caricamento dei log per la pipeline.	Stringa
Regione	Il codice per la regione in cui deve essere eseguita l'istanza Amazon EC2. Per impostazione predefinita, l'istanza viene eseguita nella stessa regione della pipeline. È possibile eseguire l'istanza nella stessa regione del set di dati dipendenti.	Enumerazione

Campi opzionali	Description	Tipo di slot
reportProgressTimeout	Timeout per chiamate successive di attività in remoto a <code>reportProgress</code> . Se impostato , le attività in remoto che non presentano avanzamenti nel periodo specificato potrebbero essere considerate bloccate ed essere quindi oggetto di un altro tentativo.	Periodo
retryDelay	La durata del timeout tra due tentativi.	Periodo
runAsUser	L'utente che deve eseguire il TaskRunner	Stringa
runsOn	Campo non consentito su questo oggetto.	Oggetto di riferimento, ad esempio, <pre>"runsOn": { "ref": "myResourceId" }</pre>

Campi opzionali	Description	Tipo di slot
scheduleType	<p>Il tipo di pianificazione consente di specificare se gli oggetti nella definizione di pipeline devono essere programmati all'inizio o alla fine dell'intervallo oppure on demand.</p> <p>I valori sono:</p> <ul style="list-style-type: none"> • <code>timeseries</code> . Le istanze sono programmate alla fine di ogni intervallo. • <code>cron</code>. Le istanze sono programmate all'inizio di ogni intervallo. • <code>ondemand</code>. Consente di eseguire una pipeline una volta per attivazione. Non è necessario clonare o ricreare la pipeline per eseguirla di nuovo. Se utilizzi una pianificazione on demand, devi specificarlo nell'oggetto predefinito e deve essere l'unico <code>scheduleType</code> specificato per gli oggetti della pipeline. Per utilizzare le pipeline on demand, chiama l'operazione <code>ActivatePipeline</code> per ogni esecuzione successiva. 	Enumerazione
securityGroupIds	Gli ID di uno o più gruppi di sicurezza Amazon EC2 da utilizzare per le istanze nel pool di risorse.	Stringa
securityGroups	Uno o più gruppi di sicurezza Amazon EC2 da utilizzare per le istanze nel pool di risorse.	Stringa
spotBidPrice	L'importo massimo per ora per la tua istanza Spot in dollari, un valore decimale compreso tra 0 e 20,00, esclusi.	Stringa
subnetId	L'ID della sottorete Amazon EC2 in cui avviare l'istanza.	Stringa

Campi opzionali	Description	Tipo di slot
<code>terminateAfter</code>	Il numero di ore dopo cui terminare la risorsa.	Periodo
<code>useOnDemandOnLastAttempt</code>	Nell'ultimo tentativo di richiesta di una risorsa Spot, effettuare una richiesta per istanze on demand invece che per istanze Spot. In questo modo, se tutti i tentativi precedenti non sono andati a buon fine, l'ultimo tentativo non viene interrotto.	Booleano
<code>workerGroup</code>	Campo non consentito su questo oggetto.	Stringa

Campi Runtime	Description	Tipo di slot
<code>@activeInstances</code>	Elenco di oggetti di istanze attive attualmente programmate.	Oggetto di riferimento, ad esempio, <code>"activeInstances": {"ref": "myRunnableObjectId"}</code>
<code>@actualEndTime</code>	L'ora in cui è terminata l'esecuzione di questo oggetto.	DateTime
<code>@actualStartTime</code>	L'ora in cui è stata avviata l'esecuzione di questo oggetto.	DateTime
<code>cancellationReason</code>	<code>cancellationReason</code> se questo oggetto è stato annullato.	Stringa
<code>@cascadeFailedOn</code>	Descrizione della catena di dipendenza che ha generato l'errore dell'oggetto.	Oggetto di riferimento, ad esempio, <code>"cascadeFailedOn": {"ref": "m</code>

Campi Runtime	Description	Tipo di slot
		yRunnable ObjectId"}
emrStepLog	I log dei passaggi sono disponibili solo per i tentativi di attività di Amazon EMR.	Stringa
errorId	ID dell'errore se l'oggetto non è riuscito.	Stringa
errorMessage	Messaggio di errore se l'oggetto non è riuscito.	Stringa
errorStackTrace	Traccia dello stack di errore se l'oggetto non è riuscito.	Stringa
@failureReason	Il motivo dell'errore della risorsa.	Stringa
@finishedTime	L'ora in cui è terminata l'esecuzione di questo oggetto.	DateTime
hadoopJobLog	I log dei lavori Hadoop sono disponibili sui tentativi di attività di Amazon EMR.	Stringa
@healthStatus	Lo stato di integrità dell'oggetto che riflette l'esito positivo o negativo dell'ultima istanza dell'oggetto che ha raggiunto lo stato di un'istanza terminata.	Stringa
@healthStatusFromInstanceid	Id dell'ultimo oggetto dell'istanza che ha raggiunto lo stato terminato.	Stringa
@ Ora healthStatusUpdated	L'ora in cui lo stato di integrità è stato aggiornato o l'ultima volta.	DateTime
hostname	Il nome host del client che si è aggiudicato il tentativo dell'attività.	Stringa
@lastDeactivatedTime	L'ora in cui l'oggetto è stato disattivato.	DateTime

Campi Runtime	Description	Tipo di slot
@latestCompletedRun Ora	L'orario dell'esecuzione più recente durante il quale l'esecuzione è stata completata.	DateTime
@latestRunTime	L'orario dell'esecuzione più recente durante il quale l'esecuzione è stata pianificata.	DateTime
@nextRunTime	L'orario dell'esecuzione da programmare come successiva.	DateTime
reportProgressTime	Il periodo di tempo più recente in cui l'attività remota ha segnalato un progresso.	DateTime
@scheduledEndTime	L'orario di termine della pianificazione per l'oggetto.	DateTime
@scheduledStartTime	L'orario di inizio della pianificazione per l'oggetto.	DateTime
@status	Lo stato di questo oggetto.	Stringa
@version	Versione della pipeline con cui l'oggetto è stato creato.	Stringa
@waitingOn	La descrizione dell'elenco di dipendenze per cui questo oggetto è in attesa.	Oggetto di riferimento, ad esempio <pre>"waitingOn": {"ref": "myRunnableObjectId"}</pre>

Campi di sistema	Description	Tipo di slot
@error	Errore che descrive il formato oggetto errato.	Stringa
@pipelineId	L'ID della pipeline a cui appartiene questo oggetto.	Stringa

Campi di sistema	Description	Tipo di slot
@sphere	La posizione di un oggetto nel ciclo di vita. I Component Objects generano Instance Objects che eseguono Attempt Objects.	Stringa

EmrCluster

Rappresenta la configurazione di un cluster Amazon EMR. Questo oggetto viene utilizzato da [EmrActivity](#) e [HadoopActivity](#) per avviare un cluster.

Indice

- [Pianificatori](#)
- [Versioni di rilascio di Amazon EMR](#)
- [Autorizzazioni Amazon EMR](#)
- [Sintassi](#)
- [Esempi](#)
- [Vedi anche](#)

Pianificatori

I pianificatori forniscono un modo per specificare l'allocazione di risorse e la prioritizzazione dei processi all'interno di un cluster Hadoop. Gli amministratori o gli utenti possono scegliere un pianificatore per diverse classi di utenti e applicazioni. Un pianificatore può usare le code per allocare risorse per utenti e applicazioni. È possibile configurare tali code quando si crea il cluster. È quindi possibile configurare priorità per determinati tipi di lavoro e utenti rispetto ad altri. In questo modo si ottiene un utilizzo efficiente delle risorse del cluster, consentendo a più utenti di inviare dati al cluster. Esistono tre tipi di pianificatori disponibili:

- [FairScheduler](#)— Tenta di pianificare le risorse in modo uniforme per un periodo di tempo significativo.
- [CapacityScheduler](#)— Utilizza le code per consentire agli amministratori del cluster di assegnare gli utenti a code con priorità e allocazione delle risorse diverse.
- Predefinito: utilizzato dal cluster, che può essere configurato dal sito.

Versioni di rilascio di Amazon EMR

Un rilascio di Amazon EMR è un insieme di applicazioni open source dell'ecosistema di big data. Ogni versione comprende diverse applicazioni, componenti e funzionalità per i big data che scegli di installare e configurare Amazon EMR quando crei un cluster. È possibile specificare la versione di rilascio con l'etichetta del rilascio. Le etichette di rilascio sono sotto forma di `emr-x.x.x`. Ad esempio, `emr-5.30.0`. I cluster Amazon EMR si basano sull'etichetta di rilascio `emr-4.0.0` e successivamente utilizzano la `releaseLabel` proprietà per specificare l'etichetta di rilascio di un oggetto. `EmrCluster` Le versioni precedenti utilizzano la proprietà `amiVersion`.

Important

Tutti i cluster Amazon EMR creati utilizzando la versione 5.22.0 o successiva utilizzano [Signature Version 4](#) per autenticare le richieste verso Amazon S3. Alcune versioni di rilascio precedenti utilizzano Signature Version 2. Il supporto di Signature Version 2 è stato interrotto. Per ulteriori informazioni, consulta [Amazon S3 Update – SigV2 Deprecation Period Extended and Modified](#). Ti consigliamo vivamente di utilizzare una versione di Amazon EMR che supporti la versione 4 di Signature. Per i rilasci di versioni precedenti, a partire da EMR 4.7.x, la versione più recente della serie è stata aggiornata per supportare Signature Version 4. Quando si utilizza una versione precedente del rilascio EMR, si consiglia di utilizzare la versione più recente della serie. Inoltre, evitare versioni precedenti a EMR 4.7.0.

Considerazioni e limitazioni

Usa la versione più recente di Task Runner

Se si utilizza un `EmrCluster` oggetto autogestito con un'etichetta di rilascio, utilizzare la versione più recente di Task Runner. Per ulteriori informazioni sui Task Runner, consulta [Utilizzo di Task Runner](#). Puoi configurare i valori delle proprietà per tutte le classificazioni di configurazione di Amazon EMR. Per ulteriori informazioni, consulta [Configurazione delle applicazioni](#) nella Amazon EMR Release Guide, [the section called “Proprietà”](#) e riferimenti [the section called “EmrConfiguration”](#) agli oggetti.

Support per iMDSv2

In precedenza, AWS Data Pipeline supportava solo iMDSv1. Ora AWS Data Pipeline supporta iMDSv2 in Amazon EMR 5.23.1, 5.27.1 e 5.32 o versioni successive e Amazon EMR 6.2 o versioni successive. iMDSv2 utilizza un metodo orientato alla sessione per gestire meglio l'autenticazione

durante il recupero delle informazioni sui metadati dalle istanze. È necessario configurare le istanze per effettuare chiamate IMDSv2 creando risorse gestite dall'utente utilizzando -2.0. TaskRunner

Amazon EMR 5.32 o versione successiva e Amazon EMR 6.x

Le serie di release di Amazon EMR 5.32 o successive e 6.x utilizzano la versione 3.x di Hadoop, che ha introdotto importanti cambiamenti nel modo in cui viene valutato il classpath di Hadoop rispetto alla versione 2.x di Hadoop. Le librerie comuni come Joda-Time sono state rimosse dal classpath.

Se [EmrActivity](#) o [HadoopActivity](#) esegue un file Jar che ha dipendenze da una libreria rimossa in Hadoop 3.x, il passaggio ha esito negativo e restituisce l'errore o. `java.lang.NoClassDefFoundError java.lang.ClassNotFoundException` Questo può accadere per i file Jar che sono stati eseguiti senza problemi utilizzando le versioni di release di Amazon EMR 5.x.

Per risolvere il problema, devi copiare le dipendenze dei file Jar nel classpath di Hadoop su un `EmrCluster` oggetto prima di avviare o il. `EmrActivity` `HadoopActivity` Forniamo uno script bash per farlo. Lo script bash è disponibile nella seguente posizione, dove si trova *MyRegion* la AWS regione in cui viene eseguito l'`EmrCluster` oggetto, ad esempio. `us-west-2`

```
s3://datapipeline-MyRegion/MyRegion/bootstrap-actions/latest/TaskRunner/copy-jars-to-hadoop-classpath.sh
```

Il modo di eseguire lo script dipende dal fatto che venga `EmrActivity` `HadoopActivity` eseguito su una risorsa gestita da AWS Data Pipeline o su una risorsa autogestita.

Se utilizzi una risorsa gestita da AWS Data Pipeline, aggiungi un `bootstrapAction` all'`EmrCluster` oggetto. `bootstrapActions` specifica lo script e i file Jar da copiare come argomenti. È possibile aggiungere fino a 255 `bootstrapAction` campi per `EmrCluster` oggetto e aggiungere un `bootstrapAction` campo a un `EmrCluster` oggetto che dispone già di azioni bootstrap.

Per specificare questo script come azione di bootstrap, usa la seguente sintassi, dove `JarFileRegion` è la regione in cui viene salvato il file Jar e ogni *MyJarFile* è il percorso assoluto in Amazon S3 di un file Jar da copiare nel classpath Hadoop. Per impostazione predefinita, non specificate i file Jar che si trovano nel classpath Hadoop.

```
s3://datapipeline-MyRegion/MyRegion/bootstrap-actions/latest/TaskRunner/copy-jars-to-hadoop-classpath.sh, JarFileRegion, MyJarFile1, MyJarFile2[, ...]
```

L'esempio seguente specifica un'azione di bootstrap che copia due file Jar in Amazon S3 `my-jar-file.jar`: e il `emr-dynamodb-tool-4.14.0-jar-with-dependencies.jar`. La regione utilizzata nell'esempio è `us-west-2`.

```
{
  "id" : "MyEmrCluster",
  "type" : "EmrCluster",
  "keyPair" : "my-key-pair",
  "masterInstanceType" : "m5.xlarge",
  "coreInstanceType" : "m5.xlarge",
  "coreInstanceCount" : "2",
  "taskInstanceType" : "m5.xlarge",
  "taskInstanceCount" : "2",
  "bootstrapAction" : ["s3://datapipeline-us-west-2/us-west-2/bootstrap-actions/latest/TaskRunner/copy-jars-to-hadoop-classpath.sh,us-west-2,s3://path/to/my-jar-file.jar,s3://dynamodb-dpl-us-west-2/emr-ddb-storage-handler/4.14.0/emr-dynamodb-tools-4.14.0-jar-with-dependencies.jar"]
}
```

È necessario salvare e attivare la pipeline affinché la modifica `bootstrapAction` alla nuova abbia effetto.

Se si utilizza una risorsa autogestita, è possibile scaricare lo script nell'istanza del cluster ed eseguirlo dalla riga di comando utilizzando SSH. Lo script crea una directory denominata `/etc/hadoop/conf/shellprofile.d` e un file denominato `datapipeline-jars.sh` in quella directory. I file jar forniti come argomenti della riga di comando vengono copiati in una directory denominata creata dallo script. `/home/hadoop/datapipeline_jars`. Se il cluster è configurato in modo diverso, modifica lo script in modo appropriato dopo averlo scaricato.

La sintassi per eseguire lo script sulla riga di comando è leggermente diversa da quella `bootstrapAction` mostrata nell'esempio precedente. Utilizzate gli spazi anziché le virgole tra gli argomenti, come illustrato nell'esempio seguente.

```
./copy-jars-to-hadoop-classpath.sh us-west-2 s3://path/to/my-jar-file.jar s3://dynamodb-dpl-us-west-2/emr-ddb-storage-handler/4.14.0/emr-dynamodb-tools-4.14.0-jar-with-dependencies.jar
```

Autorizzazioni Amazon EMR

Quando crei un ruolo IAM personalizzato, valuta attentamente le autorizzazioni minime necessarie al cluster per svolgere il suo lavoro. Assicurati di concedere l'accesso alle risorse richieste, come i

file in Amazon S3 o i dati in Amazon RDS, Amazon Redshift o DynamoDB. Se si desidera impostare `visibleToAllUsers` su `False`, il ruolo deve disporre delle autorizzazioni appropriate. Si noti che `DataPipelineDefaultRole` non dispone di tali autorizzazioni. È necessario fornire un'unione dei `DataPipelineDefaultRole` ruoli `DefaultDataPipelineResourceRole` and come ruolo `EmrCluster` oggetto oppure creare un ruolo personalizzato per questo scopo.

Sintassi

Campi Object Invocation	Description	Tipo di slot
<code>schedule</code>	<p>Questo oggetto viene richiamato entro l'esecuzione di un intervallo di pianificazione. Specificare un riferimento alla pianificazione di un altro oggetto per impostare l'ordine di esecuzione e delle dipendenze per questo oggetto. È possibile soddisfare questo requisito impostando esplicitamente una pianificazione sull'oggetto, ad esempio, specificando <code>"schedule": {"ref": "DefaultSchedule"}</code>.</p> <p>Nella maggior parte dei casi, è meglio inserire il riferimento alla pianificazione nell'oggetto pipeline di default, in modo che tutti gli oggetti possano ereditare tale pianificazione. O, se la pipeline consiste di una struttura di pianificazioni (nidificate all'interno della pianificazione principale), è possibile creare un oggetto padre che dispone di un riferimento alla pianificazione. Per ulteriori informazioni sulle configurazioni di pianificazione opzionali di esempio, consulta https://docs.aws.amazon.com/datapipeline/latest/DeveloperGuide/dp-object-schedule.html</p>	<p>Oggetto di riferimento, ad esempio <code>"schedule": {"ref": "myScheduleId"}</code></p>

Campi opzionali	Description	Tipo di slot
actionOnResourceFailure	L'operazione intrapresa dopo il fallimento di una risorsa per questa risorsa. I valori validi sono "retryall", che ritenta tutte le attività per il cluster per la durata specificata, e "retrynone".	Stringa
actionOnTaskFailure	L'operazione intrapresa dopo il fallimento dell'attività per questa risorsa. I valori validi sono "continue", ossia non terminare il cluster, e "terminate".	Stringa
additionalMasterSecurityGroupIds	Identificatore di gruppi di sicurezza master aggiuntivi del cluster EMR, che segue il modulo sg-01XXXX6a. Per ulteriori informazioni, consulta Amazon EMR Additional Security Groups nella Amazon EMR Management Guide.	Stringa
additionalSlaveSecurityGroupIds	Identificatore di gruppi di sicurezza slave aggiuntivi del cluster EMR, che segue il modulo sg-01XXXX6a.	Stringa
amiVersion	La versione di Amazon Machine Image (AMI) utilizzata da Amazon EMR per installare i nodi del cluster. Per ulteriori informazioni, consulta la Guida alla gestione di Amazon EMR .	Stringa
applications	Applicazioni da installare nel cluster con argomenti separati da virgole. Hive e Pig vengono installati per impostazione predefinita. Questo parametro è applicabile solo per Amazon EMR versione 4.0 e successive.	Stringa
attemptStatus	Lo stato segnalato più di recente dall'attività remota.	Stringa

Campi opzionali	Description	Tipo di slot
attemptTimeout	Timeout per il completamento del lavoro in remoto. Se questo campo è impostato, un'attività remota che non viene completata entro il tempo impostato di avvio viene tentata di nuovo.	Periodo
availabilityZone	La zona di disponibilità in cui eseguire il cluster.	Stringa
bootstrapAction	Un'operazione da eseguire all'avvio del cluster. Puoi specificare argomenti separati da virgole. Per specificare più operazioni, fino a 255, aggiungere più campi <code>bootstrapAction</code> . Il comportamento predefinito è avviare il cluster senza operazioni di bootstrap.	Stringa
configurazione	Configurazione per il cluster Amazon EMR. Questo parametro è applicabile solo per Amazon EMR versione 4.0 e successive.	Oggetto di riferimento, ad esempio, <pre>"configuration":{"ref":"myEmrConfigurationId"}</pre>
coreInstanceBidPrezzo	Il prezzo Spot massimo che sei disposto a pagare per le istanze Amazon EC2. Se viene specificato un prezzo di offerta, Amazon EMR utilizza le istanze Spot per il gruppo di istanze. Il prezzo è specificato in USD.	Stringa
coreInstanceCount	Numero di nodi principali da utilizzare per il cluster.	Numero intero
coreInstanceType	Il tipo di istanza Amazon EC2 da utilizzare per i nodi principali. Per informazioni, consultare Istanze Amazon EC2 supportate per i cluster Amazon EMR .	Stringa

Campi opzionali	Description	Tipo di slot
coreGroupConfigura tion	La configurazione per il gruppo di istanze principali del cluster Amazon EMR. Questo parametro è applicabile solo per Amazon EMR versione 4.0 e successive.	Oggetto di riferimen to, ad esempio "configur ation": {"ref": "myEmrCon figurationId"}
coreEbsConfiguration	La configurazione per i volumi Amazon EBS che verranno collegati a ciascuno dei nodi principali del gruppo principale del cluster Amazon EMR. Per ulteriori informazioni, consulta Tipi di istanze che supportano l'ottimizzazione di EBS nella Guida per l'utente di Amazon EC2 per le istanze Linux.	Oggetto di riferimen to, ad esempio "coreEbsC onfigurati on": {"ref": "myEbsCon figuration"}
customAmild	Si applica solo alla versione 5.7.0 e successiv e di Amazon EMR. Specifica l'ID AMI di un'AMI personalizzata da utilizzare quando Amazon EMR effettua il provisioning delle istanze Amazon EC2. Può anche essere usato al posto delle azioni di bootstrap per personalizzare le configurazioni dei nodi del cluster. Per ulteriori informazioni, consulta il seguente argomento nella Amazon EMR Management Guide. Utilizzo di un'AMI personalizzata	Stringa

Campi opzionali	Description	Tipo di slot
EbsBlockDeviceConfiguration	<p>La configurazione di un dispositivo a blocchi Amazon EBS richiesto associato al gruppo di istanze. Include un numero specificato di volumi che saranno associati a ogni istanza presente nel gruppo di istanze. Include <code>volumesPerInstance</code> e <code>volumeSpecification</code>, dove:</p> <ul style="list-style-type: none"> <code>volumesPerInstance</code> è il numero di volumi EBS con una configurazione dei volumi specifica che verrà associata a ogni istanza presente nel gruppo di istanze. <code>volumeSpecification</code> sono le specifiche e del volume Amazon EBS, come tipo di volume, IOPS e dimensione in Gigabyte (GiB), che verranno richieste per il volume EBS collegato a un'istanza EC2 nel cluster Amazon EMR. 	Oggetto di riferimento, ad esempio <code>"EbsBlockDeviceConfiguration": {"ref": "myEbsBlockDeviceConfiguration"}</code>
emrManagedMasterSecurityGroup	L'identificatore del gruppo di sicurezza principale del cluster Amazon EMR, che segue la forma <code>sg-01XXXX6a</code> . Per ulteriori informazioni, consulta Configure Security Groups nella Amazon EMR Management Guide.	Stringa
emrManagedSlaveSecurityGroup	L'identificatore del gruppo di sicurezza slave del cluster Amazon EMR, che segue il modulo <code>sg-01XXXX6a</code>	Stringa
enableDebugging	Abilita il debug sul cluster Amazon EMR.	Stringa
failureAndRerunModalities	Descrive il comportamento del nodo consumer quando le dipendenze presentano un errore o vengono di nuovo eseguite.	Enumerazione

Campi opzionali	Description	Tipo di slot
hadoopSchedulerType	Il tipo di pianificatore del cluster. I tipi validi sono: <code>PARALLEL_FAIR_SCHEDULING</code> , <code>PARALLEL_CAPACITY_SCHEDULING</code> e <code>DEFAULT_SCHEDULER</code> .	Enumerazione
httpProxy	Host proxy che i clienti utilizzano per connettersi ai servizi AWS.	Oggetto di riferimento, ad esempio «HttpProxy»: {"ref":» myHttpProxy Id "}
initTimeout	Il tempo di attesa prima dell'avvio della risorsa.	Periodo
keyPair	La coppia di chiavi Amazon EC2 da utilizzare e per accedere al nodo master del cluster Amazon EMR.	Stringa
lateAfterTimeout	Il tempo trascorso dopo l'avvio della pipeline entro il quale l'oggetto deve essere completato. Viene attivato solo quando il tipo di pianificazione non è impostato su. <code>ondemand</code>	Periodo
masterInstanceBidPrezzo	Il prezzo Spot massimo che sei disposto a pagare per le istanze Amazon EC2. Si tratta di un valore decimale compreso tra 0 e 20,00, esclusi. Il prezzo è specificato in USD. L'impostazione di questo valore abilita le istanze Spot per il nodo master del cluster Amazon EMR. Se viene specificato un prezzo di offerta, Amazon EMR utilizza le istanze Spot per il gruppo di istanze.	Stringa
masterInstanceType	Il tipo di istanza Amazon EC2 da utilizzare per il nodo master. Per informazioni, consultare Istanze Amazon EC2 supportate per i cluster Amazon EMR .	Stringa

Campi opzionali	Description	Tipo di slot
masterGroupConfiguration	La configurazione per il gruppo di istanze master del cluster Amazon EMR. Questo parametro è applicabile solo per Amazon EMR versione 4.0 e successive.	Oggetto di riferimento, ad esempio "configuration": {"ref": "myEmrConfigurationId"}
masterEbsConfiguration	La configurazione per i volumi Amazon EBS che verranno collegati a ciascuno dei nodi master del gruppo principale nel cluster Amazon EMR. Per ulteriori informazioni, consulta Tipi di istanze che supportano l'ottimizzazione di EBS nella Guida per l'utente di Amazon EC2 per le istanze Linux.	Oggetto di riferimento, ad esempio "masterEbsConfiguration": {"ref": "myEbsConfiguration"}
maxActiveInstances	Il numero massimo di istanze attive simultanee di un componente. Le riesecuzioni non contano ai fini del numero di istanze attive.	Numero intero
maximumRetries	Numero massimo di tentativi in caso di errore.	Numero intero
onFail	Un'azione da eseguire quando l'oggetto corrente ha esito negativo.	Oggetto di riferimento, ad esempio, "onFail": {"ref": "myActionId"}
onLateAction	Azioni che devono essere attivate se un oggetto non è stato ancora pianificato o non è ancora completo.	Oggetto di riferimento, ad esempio, "onLateAction": {"ref": "myActionId"}

Campi opzionali	Description	Tipo di slot
onSuccess	Un'operazione da eseguire quando l'oggetto corrente ha esito positivo.	Oggetto di riferimento, ad esempio, "onSuccess": {"ref": "myActionId"}
parent	Padre dell'oggetto corrente da cui vengono ereditati gli slot.	Oggetto di riferimento, ad esempio. "parent": {"ref": "myBaseObjectId"}
pipelineLogUri	L'URI di Amazon S3 (ad esempio 's3://BucketName/Key/ ') per caricare i log per la pipeline.	Stringa
Regione	Il codice per la regione in cui deve essere eseguito il cluster Amazon EMR. Per impostazione predefinita, il cluster viene eseguito nella stessa regione della pipeline. È possibile eseguire il cluster nella stessa regione del set di dati dipendenti.	Enumerazione
releaseLabel	Etichetta release per il cluster EMR.	Stringa
reportProgressTimeout	Timeout per chiamate successive di attività in remoto a <code>reportProgress</code> . Se impostato , le attività in remoto che non presentano avanzamenti nel periodo specificato potrebbero essere considerate bloccate e sono quindi oggetto di un altro tentativo.	Periodo
resourceRole	Il ruolo IAM AWS Data Pipeline utilizzato per creare il cluster Amazon EMR. Il ruolo predefinito è <code>DataPipelineDefaultRole</code> .	Stringa

Campi opzionali	Description	Tipo di slot
retryDelay	La durata del timeout tra due tentativi.	Periodo
role	Il ruolo IAM è passato ad Amazon EMR per creare nodi EC2.	Stringa
runsOn	Campo non consentito su questo oggetto.	Oggetto di riferimento, ad esempio, "runsOn": { "ref": "myResourceId" }
Configurazione di sicurezza	L'identificatore della configurazione di sicurezza EMR che verrà applicata al cluster. Questo parametro è applicabile solo per Amazon EMR versione 4.8.0 e successive.	Stringa
serviceAccessSecurityGroupId	L'identificatore per il gruppo di sicurezza dell'accesso al servizio del cluster Amazon EMR.	Stringa. Segue la forma sg-01XXXX6a , ad esempio, sg-1234abcd .

Campi opzionali	Description	Tipo di slot
<code>scheduleType</code>	Il tipo di pianificazione consente di specificare se gli oggetti nella definizione di pipeline devono essere programmati all'inizio o alla fine dell'intervallo. I valori sono <code>cron</code> , <code>ondemand</code> e <code>timeseries</code> . La pianificazione <code>timeseries</code> significa che le istanze sono programmate al termine di ogni intervallo. La pianificazione <code>cron</code> significa che le istanze sono programmate all'inizio di ogni intervallo. Una pianificazione <code>ondemand</code> consente di eseguire una pipeline una sola volta, per attivazione. Non è necessario clonare o ricreare la pipeline per eseguirla di nuovo. Se utilizzi una pianificazione <code>ondemand</code> , devi specificarlo nell'oggetto predefinito e deve essere l'unico <code>scheduleType</code> specificato per gli oggetti della pipeline. Per utilizzare le pipeline <code>ondemand</code> , chiama l'operazione <code>ActivatePipeline</code> per ogni esecuzione successiva.	Enumerazione
<code>subnetId</code>	L'identificatore della sottorete in cui avviare il cluster Amazon EMR.	Stringa
<code>supportedProducts</code>	Un parametro che installa software di terze parti su un cluster Amazon EMR, ad esempio una distribuzione di terze parti di Hadoop.	Stringa
<code>taskInstanceBidPrezzo</code>	Il prezzo Spot massimo che sei disposto a pagare per le istanze EC2. Un valore decimale compreso tra 0 e 20,00, esclusi. Il prezzo è specificato in USD. Se viene specificato un prezzo di offerta, Amazon EMR utilizza le istanze Spot per il gruppo di istanze.	Stringa

Campi opzionali	Description	Tipo di slot
taskInstanceCount	Il numero di nodi di attività da utilizzare per il cluster Amazon EMR.	Numero intero
taskInstanceType	Il tipo di istanza Amazon EC2 da utilizzare per i nodi di attività.	Stringa
taskGroupConfigura tion	La configurazione per il gruppo di attività del cluster Amazon EMR. Questo parametro è applicabile solo per Amazon EMR versione 4.0 e successive.	Oggetto di riferimen to, ad esempio "configur ation": {"ref": "myEmrCon figurationId"}
taskEbsConfiguration	La configurazione per i volumi Amazon EBS che verranno collegati a ciascuno dei nodi di attività del gruppo di attività nel cluster Amazon EMR. Per ulteriori informazioni, consulta Tipi di istanze che supportano l'ottimizzazione di EBS nella Guida per l'utente di Amazon EC2 per le istanze Linux.	Oggetto di riferimen to, ad esempio "taskEbsC onfigurati on": {"ref": "myEbsCon figuration"}
terminateAfter	Termina la risorsa dopo queste numerose ore.	Numero intero

Campi opzionali	Description	Tipo di slot
VolumeSpecification	<p>Le specifiche del volume di Amazon EBS, come tipo di volume, IOPS e dimensione in Gigabyte (GiB), che verranno richieste per il volume Amazon EBS collegato a un'istanza Amazon EC2 nel cluster Amazon EMR. Il nodo può essere principale, master o di task.</p> <p>VolumeSpecification include:</p> <ul style="list-style-type: none"> • <code>iops()</code> intero. Il numero di operazioni di I/O al secondo (IOPS) che il volume Amazon EBS supporta, ad esempio 1000. Per ulteriori informazioni, consulta le caratteristiche di I/O di EBS nella Guida per l'utente di Amazon EC2 per le istanze Linux. • <code>sizeinGB()</code> . Numero intero. La dimensione del volume Amazon EBS, in gibibyte (GiB), ad esempio 500. Per informazioni sulle combinazioni valide di tipi di volume e dimensioni dei dischi rigidi, consulta EBS Volume Types nella Amazon EC2 User Guide for Linux Instances. • <code>volumeType</code> . Stringa. Il tipo di volume Amazon EBS, ad esempio gp2. I tipi di volume supportati includono standard, gp2, io1, st1, sc1 e molti altri. Per ulteriori informazioni, consulta EBS Volume Types nella Amazon EC2 User Guide for Linux Instances. 	<p>Oggetto di riferimento, ad esempio</p> <pre> "VolumeSpecification": {"ref": "myVolumeSpecification"} </pre>
useOnDemandOnLastAttempt	<p>Nell'ultimo tentativo di richiesta di una risorsa, effettuare una richiesta per istanze on demand invece che per istanze Spot. In questo modo, se tutti i tentativi precedenti non sono andati a buon fine, l'ultimo tentativo non viene interrotto.</p>	<p>Booleano</p>

Campi opzionali	Description	Tipo di slot
workerGroup	Campo non consentito su questo oggetto.	Stringa
Campi Runtime	Description	Tipo di slot
@activeInstances	Elenco di oggetti di istanze attive attualmente programmate.	Oggetto di riferimento, ad esempio «activeInstances»: {"ref":» Id "myRunnableObject
@actualEndTime	L'ora in cui è terminata l'esecuzione di questo oggetto.	DateTime
@actualStartTime	L'ora in cui è stata avviata l'esecuzione di questo oggetto.	DateTime
cancellationReason	CancellationReason se questo oggetto è stato annullato.	Stringa
@cascadeFailedOn	Descrizione della catena di dipendenza che ha generato l'errore dell'oggetto.	Oggetto di riferimento, ad esempio "cascadeFailedOn«: {"ref":» myRunnableObject Id "}
emrStepLog	I log dei passaggi sono disponibili solo per i tentativi di attività di Amazon EMR.	Stringa
errorId	ID dell'errore se l'oggetto non è riuscito.	Stringa
errorMessage	Messaggio di errore se l'oggetto non è riuscito.	Stringa
errorStackTrace	Traccia dello stack di errore se l'oggetto non è riuscito.	Stringa
@failureReason	Il motivo dell'errore della risorsa.	Stringa

Campi Runtime	Description	Tipo di slot
@finishedTime	L'ora in cui è terminata l'esecuzione di questo oggetto.	DateTime
hadoopJobLog	I log dei lavori Hadoop sono disponibili sui tentativi di attività di Amazon EMR.	Stringa
@healthStatus	Lo stato di integrità dell'oggetto che riflette l'esito positivo o negativo dell'ultima istanza dell'oggetto che ha raggiunto lo stato di un'istanza terminata.	Stringa
@healthStatusFromInstanceid	Id dell'ultimo oggetto dell'istanza che ha raggiunto lo stato terminato.	Stringa
@ Ora healthStatusUpdated	L'ora in cui lo stato di integrità è stato aggiornato o l'ultima volta.	DateTime
hostname	Il nome host del client che si è aggiudicato il tentativo dell'attività.	Stringa
@lastDeactivatedTime	L'ora in cui l'oggetto è stato disattivato.	DateTime
@ latestCompletedRun Ora	L'orario dell'esecuzione più recente durante il quale l'esecuzione è stata completata.	DateTime
@latestRunTime	L'orario dell'esecuzione più recente durante il quale l'esecuzione è stata pianificata.	DateTime
@nextRunTime	L'orario dell'esecuzione da programmare come successiva.	DateTime
reportProgressTime	Il periodo di tempo più recente in cui l'attività remota ha segnalato un progresso.	DateTime
@scheduledEndTime	L'orario di termine della pianificazione per un oggetto.	DateTime

Campi Runtime	Description	Tipo di slot
@scheduledStartTime	L'orario di inizio della pianificazione per l'oggetto.	DateTime
@status	Lo stato di questo oggetto.	Stringa
@version	Versione della pipeline con cui l'oggetto è stato creato.	Stringa
@waitingOn	La descrizione dell'elenco di dipendenze per cui questo oggetto è in attesa.	Oggetto di riferimento, ad esempio «waitingOn»: {"ref":» myRunnableObject Id "}

Campi di sistema	Description	Tipo di slot
@error	Errore che descrive il formato oggetto errato.	Stringa
@pipelineId	L'ID della pipeline a cui appartiene questo oggetto.	Stringa
@sphere	La posizione di un oggetto nel ciclo di vita. I Component Objects generano Instance Objects che eseguono Attempt Objects.	Stringa

Esempi

I seguenti sono esempi di questo tipo di oggetto.

Indice

- [Avvia un cluster Amazon EMR con HadoopVersion](#)
- [Avvia un cluster Amazon EMR con etichetta di rilascio emr-4.x o successiva](#)
- [Installa software aggiuntivo sul tuo cluster Amazon EMR](#)
- [Disabilitare la crittografia lato server sulle versioni 3.x](#)

- [Disabilitare la crittografia lato server sulle versioni 4.x](#)
- [Configurare Hadoop KMS ACL e creare zone di crittografia in HDFS](#)
- [Specificare ruoli IAM personalizzati](#)
- [Usa EmrCluster la risorsa nell'SDK AWS per Java](#)
- [Configurazione di un cluster Amazon EMR in una sottorete privata](#)
- [Collegare i volumi EBS ai nodi del cluster](#)

Avvia un cluster Amazon EMR con HadoopVersion

Example

L'esempio seguente avvia un cluster Amazon EMR utilizzando AMI versione 1.0 e Hadoop 0.20.

```
{
  "id" : "MyEmrCluster",
  "type" : "EmrCluster",
  "hadoopVersion" : "0.20",
  "keyPair" : "my-key-pair",
  "masterInstanceType" : "m3.xlarge",
  "coreInstanceType" : "m3.xlarge",
  "coreInstanceCount" : "10",
  "taskInstanceType" : "m3.xlarge",
  "taskInstanceCount" : "10",
  "bootstrapAction" : ["s3://Region.elasticmapreduce/bootstrap-actions/configure-hadoop,arg1,arg2,arg3","s3://Region.elasticmapreduce/bootstrap-actions/configure-hadoop/configure-other-stuff,arg1,arg2"]
}
```

Avvia un cluster Amazon EMR con etichetta di rilascio emr-4.x o successiva

Example

L'esempio seguente avvia un cluster Amazon EMR utilizzando il `releaseLabel` campo più recente:

```
{
  "id" : "MyEmrCluster",
  "type" : "EmrCluster",
  "keyPair" : "my-key-pair",
  "masterInstanceType" : "m3.xlarge",
```

```

"coreInstanceType" : "m3.xlarge",
"coreInstanceCount" : "10",
"taskInstanceType" : "m3.xlarge",
"taskInstanceCount": "10",
"releaseLabel": "emr-4.1.0",
"applications": ["spark", "hive", "pig"],
"configuration": {"ref": "myConfiguration"}
}

```

Installa software aggiuntivo sul tuo cluster Amazon EMR

Example

EmrCluster fornisce il `supportedProducts` campo che installa software di terze parti su un cluster Amazon EMR, ad esempio consente di installare una distribuzione personalizzata di Hadoop, come MapR. Accetta un elenco separato da virgole di argomenti per il software di terze parti da leggere e in base al quale agire. L'esempio seguente mostra come usare il campo `supportedProducts` di `EmrCluster` per creare un cluster personalizzato edizione MapR M3 con Karmasphere Analytics installato ed eseguire un oggetto `EmrActivity` su di esso.

```

{
  "id": "MyEmrActivity",
  "type": "EmrActivity",
  "schedule": {"ref": "ResourcePeriod"},
  "runsOn": {"ref": "MyEmrCluster"},
  "postStepCommand": "echo Ending job >> /mnt/var/log/stepCommand.txt",
  "preStepCommand": "echo Starting job > /mnt/var/log/stepCommand.txt",
  "step": "/home/hadoop/contrib/streaming/hadoop-streaming.jar, -input, s3n://
elasticmapreduce/samples/wordcount/input, -output, \
  hdfs:///output32113/, -mapper, s3n://elasticmapreduce/samples/wordcount/
wordSplitter.py, -reducer, aggregate"
},
{
  "id": "MyEmrCluster",
  "type": "EmrCluster",
  "schedule": {"ref": "ResourcePeriod"},
  "supportedProducts": ["mapr, --edition, m3, --version, 1.2, --key1, value1", "karmasphere-
enterprise-utility"],
  "masterInstanceType": "m3.xlarge",
  "taskInstanceType": "m3.xlarge"
}

```

Disabilitare la crittografia lato server sulle versioni 3.x

Example

Un'attività `EmrCluster` con una versione Hadoop 2.x creata da AWS Data Pipeline abilita la crittografia lato server per impostazione predefinita. Se si desidera disattivare la crittografia lato server, è necessario specificare un'operazione di bootstrap nella definizione di un oggetto del cluster.

L'esempio seguente crea un'attività `EmrCluster` con la crittografia lato server disabilitata:

```
{
  "id": "NoSSEmrCluster",
  "type": "EmrCluster",
  "hadoopVersion": "2.x",
  "keyPair": "my-key-pair",
  "masterInstanceType": "m3.xlarge",
  "coreInstanceType": "m3.large",
  "coreInstanceCount": "10",
  "taskInstanceType": "m3.large",
  "taskInstanceCount": "10",
  "bootstrapAction": ["s3://Region.elasticmapreduce/bootstrap-actions/configure-hadoop,-e,fs.s3.enableServerSideEncryption=false"]
}
```

Disabilitare la crittografia lato server sulle versioni 4.x

Example

È necessario disabilitare la crittografia lato server utilizzando un oggetto `EmrConfiguration`.

L'esempio seguente crea un'attività `EmrCluster` con la crittografia lato server disabilitata:

```
{
  "name": "ReleaseLabelCluster",
  "releaseLabel": "emr-4.1.0",
  "applications": ["spark", "hive", "pig"],
  "id": "myResourceId",
  "type": "EmrCluster",
  "configuration": {
    "ref": "disableSSE"
  }
},
{
  "name": "disableSSE",
```

```

    "id": "disableSSE",
    "type": "EmrConfiguration",
    "classification": "emrfs-site",
    "property": [{
      "ref": "enableServerSideEncryption"
    }
  ],
  {
    "name": "enableServerSideEncryption",
    "id": "enableServerSideEncryption",
    "type": "Property",
    "key": "fs.s3.enableServerSideEncryption",
    "value": "false"
  }
}

```

Configurare Hadoop KMS ACL e creare zone di crittografia in HDFS

Example

I seguenti oggetti creano ACL per Hadoop KMS e creano zone di crittografia e le chiavi di crittografia corrispondenti in HDFS:

```

{
  "name": "kmsAcls",
  "id": "kmsAcls",
  "type": "EmrConfiguration",
  "classification": "hadoop-kms-acls",
  "property": [
    {"ref": "kmsBlacklist"},
    {"ref": "kmsAcl"}
  ]
},
{
  "name": "hdfsEncryptionZone",
  "id": "hdfsEncryptionZone",
  "type": "EmrConfiguration",
  "classification": "hdfs-encryption-zones",
  "property": [
    {"ref": "hdfsPath1"},
    {"ref": "hdfsPath2"}
  ]
},
{

```

```

    "name": "kmsBlacklist",
    "id": "kmsBlacklist",
    "type": "Property",
    "key": "hadoop.kms.blacklist.CREATE",
    "value": "foo,myBannedUser"
  },
  {
    "name": "kmsAcl",
    "id": "kmsAcl",
    "type": "Property",
    "key": "hadoop.kms.acl.ROLLOVER",
    "value": "myAllowedUser"
  },
  {
    "name": "hdfsPath1",
    "id": "hdfsPath1",
    "type": "Property",
    "key": "/myHDFSPath1",
    "value": "path1_key"
  },
  {
    "name": "hdfsPath2",
    "id": "hdfsPath2",
    "type": "Property",
    "key": "/myHDFSPath2",
    "value": "path2_key"
  }
}

```

Specificare ruoli IAM personalizzati

Example

Per impostazione predefinita, AWS Data Pipeline passa `DataPipelineDefaultRole` come ruolo del servizio Amazon EMR e `DataPipelineDefaultResourceRole` come profilo dell'istanza Amazon EC2 per creare risorse per tuo conto. Tuttavia, puoi creare un ruolo di servizio Amazon EMR personalizzato e un profilo di istanza personalizzato e utilizzarli al loro posto. AWS Data Pipeline deve disporre di autorizzazioni sufficienti per creare cluster utilizzando il ruolo personalizzato e deve essere aggiunto AWS Data Pipeline come entità attendibile.

L'oggetto di esempio seguente specifica i ruoli personalizzati per il cluster Amazon EMR:

```

{
  "id": "MyEmrCluster",

```

```
"type": "EmrCluster",
"hadoopVersion": "2.x",
"keyPair": "my-key-pair",
"masterInstanceType": "m3.xlarge",
"coreInstanceType": "m3.large",
"coreInstanceCount": "10",
"taskInstanceType": "m3.large",
"taskInstanceCount": "10",
"role": "emrServiceRole",
"resourceRole": "emrInstanceProfile"
}
```

Usa `EmrCluster` la risorsa nell'SDK AWS per Java

Example

L'esempio seguente mostra come utilizzare un `EmrCluster` e `EmrActivity` creare un cluster Amazon EMR 4.x per eseguire una fase Spark utilizzando Java SDK:

```
public class dataPipelineEmr4 {

    public static void main(String[] args) {

        AWSCredentials credentials = null;
        credentials = new ProfileCredentialsProvider("/path/to/
        AwsCredentials.properties","default").getCredentials();
        DataPipelineClient dp = new DataPipelineClient(credentials);
        CreatePipelineRequest createPipeline = new
        CreatePipelineRequest().withName("EMR4SDK").withUniqueId("unique");
        CreatePipelineResult createPipelineResult = dp.createPipeline(createPipeline);
        String pipelineId = createPipelineResult.getPipelineId();

        PipelineObject emrCluster = new PipelineObject()
            .withName("EmrClusterObj")
            .withId("EmrClusterObj")
            .withFields(
                new Field().withKey("releaseLabel").withStringValue("emr-4.1.0"),
                new Field().withKey("coreInstanceCount").withStringValue("3"),
                new Field().withKey("applications").withStringValue("spark"),
                new Field().withKey("applications").withStringValue("Presto-Sandbox"),
                new Field().withKey("type").withStringValue("EmrCluster"),
                new Field().withKey("keyPair").withStringValue("myKeyName"),
                new Field().withKey("masterInstanceType").withStringValue("m3.xlarge"),
```

```
new Field().withKey("coreInstanceType").withStringValue("m3.xlarge")
);

PipelineObject emrActivity = new PipelineObject()
    .withName("EmrActivityObj")
    .withId("EmrActivityObj")
    .withFields(
        new Field().withKey("step").withStringValue("command-runner.jar,spark-submit,--
executor-memory,1g,--class,org.apache.spark.examples.SparkPi,/usr/lib/spark/lib/spark-
examples.jar,10"),
        new Field().withKey("runsOn").withRefValue("EmrClusterObj"),
        new Field().withKey("type").withStringValue("EmrActivity")
    );

PipelineObject schedule = new PipelineObject()
    .withName("Every 15 Minutes")
    .withId("DefaultSchedule")
    .withFields(
        new Field().withKey("type").withStringValue("Schedule"),
        new Field().withKey("period").withStringValue("15 Minutes"),
        new Field().withKey("startAt").withStringValue("FIRST_ACTIVATION_DATE_TIME")
    );

PipelineObject defaultObject = new PipelineObject()
    .withName("Default")
    .withId("Default")
    .withFields(
        new Field().withKey("failureAndRerunMode").withStringValue("CASCADE"),
        new Field().withKey("schedule").withRefValue("DefaultSchedule"),
        new
Field().withKey("resourceRole").withStringValue("DataPipelineDefaultResourceRole"),
        new Field().withKey("role").withStringValue("DataPipelineDefaultRole"),
        new Field().withKey("pipelineLogUri").withStringValue("s3://myLogUri"),
        new Field().withKey("scheduleType").withStringValue("cron")
    );

List<PipelineObject> pipelineObjects = new ArrayList<PipelineObject>();

pipelineObjects.add(emrActivity);
pipelineObjects.add(emrCluster);
pipelineObjects.add(defaultObject);
pipelineObjects.add(schedule);

PutPipelineDefinitionRequest putPipelineDefintion = new PutPipelineDefinitionRequest()
```

```

        .withPipelineId(pipelineId)
        .withPipelineObjects(pipelineObjects);

PutPipelineDefinitionResult putPipelineResult =
dp.putPipelineDefinition(putPipelineDefintion);
System.out.println(putPipelineResult);

ActivatePipelineRequest activatePipelineReq = new ActivatePipelineRequest()
    .withPipelineId(pipelineId);
ActivatePipelineResult activatePipelineRes = dp.activatePipeline(activatePipelineReq);

    System.out.println(activatePipelineRes);
    System.out.println(pipelineId);

}

}

```

Configurazione di un cluster Amazon EMR in una sottorete privata

Example

Questo esempio include una configurazione che avvia il cluster in una sottorete privata in un VPC. Per ulteriori informazioni, consulta [Launch Amazon EMR Clusters in un VPC](#) nella Amazon EMR Management Guide. Questa configurazione è opzionale. È possibile utilizzarla in qualsiasi pipeline che utilizza un oggetto `EmrCluster`.

Per avviare un cluster Amazon EMR in una sottorete privata, specifica `SubnetId` «,» e `serviceAccessSecurityGroupId` nella `emrManagedSlaveSecurityGroupId` tua configurazione. `emrManagedMasterSecurityGroupId` `EmrCluster`

```

{
  "objects": [
    {
      "output": {
        "ref": "S3BackupLocation"
      },
      "input": {
        "ref": "DDBSourceTable"
      },
      "maximumRetries": "2",
      "name": "TableBackupActivity",

```

```

    "step": "s3://dynamodb-emr-#{myDDBRegion}/emr-ddb-storage-handler/2.1.0/emr-
ddb-2.1.0.jar,org.apache.hadoop.dynamodb.tools.DynamoDbExport,#{output.directoryPath},#{input.t
    "id": "TableBackupActivity",
    "runsOn": {
      "ref": "EmrClusterForBackup"
    },
    "type": "EmrActivity",
    "resizeClusterBeforeRunning": "false"
  },
  {
    "readThroughputPercent": "#{myDDBReadThroughputRatio}",
    "name": "DDBSourceTable",
    "id": "DDBSourceTable",
    "type": "DynamoDBDataNode",
    "tableName": "#{myDDBTableName}"
  },
  {
    "directoryPath": "#{myOutputS3Loc}/#{format(@scheduledStartTime, 'YYYY-MM-dd-HH-
mm-ss')}",
    "name": "S3BackupLocation",
    "id": "S3BackupLocation",
    "type": "S3DataNode"
  },
  {
    "name": "EmrClusterForBackup",
    "coreInstanceCount": "1",
    "taskInstanceCount": "1",
    "taskInstanceType": "m4.xlarge",
    "coreInstanceType": "m4.xlarge",
    "releaseLabel": "emr-4.7.0",
    "masterInstanceType": "m4.xlarge",
    "id": "EmrClusterForBackup",
    "subnetId": "#{mySubnetId}",
    "emrManagedMasterSecurityGroupId": "#{myMasterSecurityGroup}",
    "emrManagedSlaveSecurityGroupId": "#{mySlaveSecurityGroup}",
    "serviceAccessSecurityGroupId": "#{myServiceAccessSecurityGroup}",
    "region": "#{myDDBRegion}",
    "type": "EmrCluster",
    "keyPair": "user-key-pair"
  },
  {
    "failureAndRerunMode": "CASCADE",
    "resourceRole": "DataPipelineDefaultResourceRole",
    "role": "DataPipelineDefaultRole",

```

```

    "pipelineLogUri": "#{myPipelineLogUri}",
    "scheduleType": "ONDEMAND",
    "name": "Default",
    "id": "Default"
  }
],
"parameters": [
  {
    "description": "Output S3 folder",
    "id": "myOutputS3Loc",
    "type": "AWS::S3::ObjectKey"
  },
  {
    "description": "Source DynamoDB table name",
    "id": "myDDBTableName",
    "type": "String"
  },
  {
    "default": "0.25",
    "watermark": "Enter value between 0.1-1.0",
    "description": "DynamoDB read throughput ratio",
    "id": "myDDBReadThroughputRatio",
    "type": "Double"
  },
  {
    "default": "us-east-1",
    "watermark": "us-east-1",
    "description": "Region of the DynamoDB table",
    "id": "myDDBRegion",
    "type": "String"
  }
],
"values": {
  "myDDBRegion": "us-east-1",
  "myDDBTableName": "ddb_table",
  "myDDBReadThroughputRatio": "0.25",
  "myOutputS3Loc": "s3://s3_path",
  "mySubnetId": "subnet_id",
  "myServiceAccessSecurityGroup": "service access security group",
  "mySlaveSecurityGroup": "slave security group",
  "myMasterSecurityGroup": "master security group",
  "myPipelineLogUri": "s3://s3_path"
}
}

```

Collegare i volumi EBS ai nodi del cluster

Example

È possibile allegare i volumi EBS a qualsiasi tipo di nodo nel cluster EMR all'interno della pipeline. Per allegare volumi EBS ai nodi, utilizzare `coreEbsConfiguration`, `masterEbsConfiguration` e `TaskEbsConfiguration` nella configurazione `EmrCluster`.

Questo esempio di cluster Amazon EMR utilizza i volumi Amazon EBS per i nodi master, task e core. Per ulteriori informazioni, consulta [i volumi Amazon EBS in Amazon EMR](#) nella Amazon EMR Management Guide.

Queste configurazioni sono opzionali. È possibile utilizzarle in qualsiasi pipeline che utilizza un oggetto `EmrCluster`.

Nella pipeline, fare clic sulla configurazione dell'oggetto `EmrCluster`, scegliere Master EBS Configuration (Configurazione Master EBS), Core EBS Configuration (Configurazione EBS Core) o Task EBS Configuration (Configurazione EBS Task), quindi immettere i dettagli di configurazione simili a quello dell'esempio seguente.

```
{
  "objects": [
    {
      "output": {
        "ref": "S3BackupLocation"
      },
      "input": {
        "ref": "DDBSourceTable"
      },
      "maximumRetries": "2",
      "name": "TableBackupActivity",
      "step": "s3://dynamodb-emr-#{myDDBRegion}/emr-ddb-storage-handler/2.1.0/emr-ddb-2.1.0.jar,org.apache.hadoop.dynamodb.tools.DynamoDbExport,#{output.directoryPath},#{input.t",
      "id": "TableBackupActivity",
      "runsOn": {
        "ref": "EmrClusterForBackup"
      },
      "type": "EmrActivity",
      "resizeClusterBeforeRunning": "false"
    },
  ],
}
```

```

    "readThroughputPercent": "#{myDDBReadThroughputRatio}",
    "name": "DDBSourceTable",
    "id": "DDBSourceTable",
    "type": "DynamoDBDataNode",
    "tableName": "#{myDDBTableName}"
  },
  {
    "directoryPath": "#{myOutputS3Loc}/#{format(@scheduledStartTime, 'YYYY-MM-dd-HH-mm-ss')}",
    "name": "S3BackupLocation",
    "id": "S3BackupLocation",
    "type": "S3DataNode"
  },
  {
    "name": "EmrClusterForBackup",
    "coreInstanceCount": "1",
    "taskInstanceCount": "1",
    "taskInstanceType": "m4.xlarge",
    "coreInstanceType": "m4.xlarge",
    "releaseLabel": "emr-4.7.0",
    "masterInstanceType": "m4.xlarge",
    "id": "EmrClusterForBackup",
    "subnetId": "#{mySubnetId}",
    "emrManagedMasterSecurityGroupId": "#{myMasterSecurityGroup}",
    "emrManagedSlaveSecurityGroupId": "#{mySlaveSecurityGroup}",
    "region": "#{myDDBRegion}",
    "type": "EmrCluster",
    "coreEbsConfiguration": {
      "ref": "EBSConfiguration"
    },
    "masterEbsConfiguration": {
      "ref": "EBSConfiguration"
    },
    "taskEbsConfiguration": {
      "ref": "EBSConfiguration"
    },
    "keyPair": "user-key-pair"
  },
  {
    "name": "EBSConfiguration",
    "id": "EBSConfiguration",
    "ebsOptimized": "true",
    "ebsBlockDeviceConfig" : [
      { "ref": "EbsBlockDeviceConfig" }
    ]
  }

```

```
    ],
    "type": "EbsConfiguration"
  },
  {
    "name": "EbsBlockDeviceConfig",
    "id": "EbsBlockDeviceConfig",
    "type": "EbsBlockDeviceConfig",
    "volumesPerInstance" : "2",
    "volumeSpecification" : {
      "ref": "VolumeSpecification"
    }
  },
  {
    "name": "VolumeSpecification",
    "id": "VolumeSpecification",
    "type": "VolumeSpecification",
    "sizeInGB": "500",
    "volumeType": "io1",
    "iops": "1000"
  },
  {
    "failureAndRerunMode": "CASCADE",
    "resourceRole": "DataPipelineDefaultResourceRole",
    "role": "DataPipelineDefaultRole",
    "pipelineLogUri": "#{myPipelineLogUri}",
    "scheduleType": "ONDEMAND",
    "name": "Default",
    "id": "Default"
  }
],
"parameters": [
  {
    "description": "Output S3 folder",
    "id": "myOutputS3Loc",
    "type": "AWS::S3::ObjectKey"
  },
  {
    "description": "Source DynamoDB table name",
    "id": "myDDBTableName",
    "type": "String"
  },
  {
    "default": "0.25",
    "watermark": "Enter value between 0.1-1.0",
```

```
    "description": "DynamoDB read throughput ratio",
    "id": "myDDBReadThroughputRatio",
    "type": "Double"
  },
  {
    "default": "us-east-1",
    "watermark": "us-east-1",
    "description": "Region of the DynamoDB table",
    "id": "myDDBRegion",
    "type": "String"
  }
],
"values": {
  "myDDBRegion": "us-east-1",
  "myDDBTableName": "ddb_table",
  "myDDBReadThroughputRatio": "0.25",
  "myOutputS3Loc": "s3://s3_path",
  "mySubnetId": "subnet_id",
  "mySlaveSecurityGroup": "slave security group",
  "myMasterSecurityGroup": "master security group",
  "myPipelineLogUri": "s3://s3_path"
}
}
```

Vedi anche

- [EmrActivity](#)

HttpProxy

HttpProxy consente di configurare il proprio proxy e fare in modo che Task Runner acceda al AWS Data Pipeline servizio tramite esso. Non è necessario configurare un Task Runner in esecuzione con queste informazioni.

Esempio di un in HttpProxy TaskRunner

La seguente definizione di pipeline mostra un oggetto HttpProxy:

```
{
  "objects": [
    {
      "schedule": {
```

```
    "ref": "Once"
  },
  "pipelineLogUri": "s3://myDPLogUri/path",
  "name": "Default",
  "id": "Default"
},
{
  "name": "test_proxy",
  "hostname": "hostname",
  "port": "port",
  "username": "username",
  "*password": "password",
  "windowsDomain": "windowsDomain",
  "type": "HttpProxy",
  "id": "test_proxy",
},
{
  "name": "ShellCommand",
  "id": "ShellCommand",
  "runsOn": {
    "ref": "Resource"
  },
  "type": "ShellCommandActivity",
  "command": "echo 'hello world' "
},
{
  "period": "1 day",
  "startDateTime": "2013-03-09T00:00:00",
  "name": "Once",
  "id": "Once",
  "endDateTime": "2013-03-10T00:00:00",
  "type": "Schedule"
},
{
  "role": "dataPipelineRole",
  "httpProxy": {
    "ref": "test_proxy"
  },
  "actionOnResourceFailure": "retrynone",
  "maximumRetries": "0",
  "type": "Ec2Resource",
  "terminateAfter": "10 minutes",
  "resourceRole": "resourceRole",
  "name": "Resource",
```

```

    "actionOnTaskFailure": "terminate",
    "securityGroups": "securityGroups",
    "keyPair": "keyPair",
    "id": "Resource",
    "region": "us-east-1"
  }
],
"parameters": []
}

```

Sintassi

Campi obbligatori	Description	Tipo di slot
hostname	Hosting del proxy che i clienti possano utilizzare e per connettersi ai servizi AWS.	Stringa
port	Porta dell'host proxy che i clienti possano utilizzare per connettersi ai Servizi AWS.	Stringa

Campi opzionali	Description	Tipo di slot
parent	Padre dell'oggetto corrente da cui saranno ereditati gli slot.	Oggetto di riferimento, ad esempio «parent»: {"ref":» myBaseObject Id "}
*password	Password per il proxy.	Stringa
s3 NoProxy	Disabilita il proxy HTTP durante la connessione ad Amazon S3	Booleano
username	Nome utente per il proxy	Stringa
windowsDomain	Il nome del dominio di Windows per il proxy NTLM.	Stringa

Campi opzionali	Description	Tipo di slot
windowsWorkgroup	Il nome del gruppo di lavoro di Windows per il proxy NTLM.	Stringa

Campi Runtime	Description	Tipo di slot
@version	Versione della pipeline con cui l'oggetto è stato creato.	Stringa

Campi di sistema	Description	Tipo di slot
@error	Errore che descrive il formato oggetto errato.	Stringa
@pipelineId	L'id della pipeline a cui appartiene questo oggetto.	Stringa
@sphere	La sfera di un oggetto indica la propria posizione nel ciclo di vita: i Component Objects generano Instance Objects che eseguono Attempt Objects.	Stringa

Precondizioni

Di seguito sono elencati gli oggetti di precondizioni AWS Data Pipeline:

Oggetti

- [DynamoDB DataExists](#)
- [DynamoDB TableExists](#)
- [Exists](#)
- [S3 KeyExists](#)
- [S3 PrefixNotEmpty](#)

- [ShellCommandPrecondition](#)

DynamoDB DataExists

Un prerequisito per verificare l'esistenza di dati in una tabella DynamoDB.

Sintassi

Campi obbligatori	Description	Tipo di slot
role	Specifica il ruolo da utilizzare per eseguire la preconditione.	Stringa
tableName	Tabella DynamoDB da verificare.	Stringa

Campi opzionali	Description	Tipo di slot
attemptStatus	Lo stato segnalato più di recente dall'attività remota.	Stringa
attemptTimeout	Timeout per il completamento del lavoro in remoto. Se questo campo è impostato, un'attività remota che non viene completata entro il tempo impostato di avvio viene tentata di nuovo.	Periodo
failureAndRerunModo	Descrive il comportamento del nodo consumer quando le dipendenze presentano un errore o vengono di nuovo eseguite	Enumerazione
lateAfterTimeout	Il tempo trascorso dopo l'inizio della pipeline entro il quale l'oggetto deve essere completato. Viene attivato solo quando il tipo di pianificazione non è impostato su. ondemand	Periodo
maximumRetries	Numero massimo di tentativi in caso di errore	Numero intero

Campi opzionali	Description	Tipo di slot
onFail	Un'azione da eseguire quando l'oggetto corrente ha esito negativo.	Oggetto di riferimento, ad esempio «onFail»: {"ref":» myActionId «}
onLateAction	Azioni che devono essere attivate se un oggetto non è stato ancora pianificato o non è ancora completo.	Oggetto di riferimento, ad esempio "onLateAction«: {"ref":» myActionId «}
onSuccess	Un'operazione da eseguire quando l'oggetto corrente ha esito positivo.	Oggetto di riferimento, ad esempio «onSuccess»: {"ref":» myActionId «}
parent	Padre dell'oggetto corrente da cui saranno ereditati gli slot.	Oggetto di riferimento, ad esempio «parent»: {"ref":» myBaseObject Id "}
preconditionTimeout	Il periodo dall'inizio dopo il quale la condizione viene contrassegnata come non riuscita se ancora non è stata soddisfatta.	Periodo
reportProgressTimeout	Timeout per chiamate successive di attività in remoto a reportProgress. Se impostato, le attività in remoto che non presentano avanzamenti nel periodo specificato potrebbero essere considerate bloccate e sono quindi oggetto di un altro tentativo.	Periodo
retryDelay	La durata del timeout tra due tentativi.	Periodo

Campi Runtime	Description	Tipo di slot
@activeInstances	Elenco di oggetti di istanze attive attualmente programmate.	Oggetto di riferimento, ad esempio «activeInstances»: {"ref":» myRunnableObject Id "}
@actualEndTime	L'ora in cui è terminata l'esecuzione di questo oggetto.	DateTime
@actualStartTime	L'ora in cui è stata avviata l'esecuzione di questo oggetto.	DateTime
cancellationReason	CancellationReason se questo oggetto è stato annullato.	Stringa
@cascadeFailedOn	Descrizione della catena di dipendenza che ha generato l'errore dell'oggetto.	Oggetto di riferimento, ad esempio "cascadeFailedOn«: {" ref":» myRunnabl eObject Id "}
currentRetryCount	Numero di volte in cui la preconditione è stato provata in questo tentativo.	Stringa
emrStepLog	Log della fase EMR disponibili solo sui tentativi delle attività EMR	Stringa
errorId	ErrorId se l'oggetto non è riuscito.	Stringa
errorMessage	ErrorMessage se l'oggetto non è riuscito.	Stringa
errorStackTrace	Traccia dello stack di errore se l'oggetto non è riuscito.	Stringa
hadoopJobLog	Log delle attività Hadoop disponibili per le attività basate su EMR.	Stringa

Campi Runtime	Description	Tipo di slot
hostname	Il nome host del client che si è aggiudicato il tentativo dell'attività.	Stringa
lastRetryTime	L'ultima volta in cui la precondizione è stato provata all'interno di questo tentativo.	Stringa
nodo	Il nodo per il quale viene eseguita questa precondizione	Oggetto di riferimento, ad esempio «node»: {"ref":» myRunnabl eObject Id "}
reportProgressTime	Il periodo di tempo più recente in cui l'attività remota ha segnalato un progresso.	DateTime
@scheduledEndTime	L'orario di termine della pianificazione per un oggetto.	DateTime
@scheduledStartTime	L'orario di inizio della pianificazione per l'oggetto.	DateTime
@status	Lo stato di questo oggetto.	Stringa
@version	Versione della pipeline con cui l'oggetto è stato creato.	Stringa
@waitingOn	Descrizione dell'elenco di dipendenze per cui questo oggetto è in attesa.	Oggetto di riferimen to, ad esempio «waitingOn»: {"ref":» myRunnableObject Id "}

Campi di sistema	Description	Tipo di slot
@error	Errore che descrive il formato oggetto errato.	Stringa

Campi di sistema	Description	Tipo di slot
@pipelineId	L'id della pipeline a cui appartiene questo oggetto.	Stringa
@sphere	La sfera di un oggetto indica la propria posizione nel ciclo di vita: i Component Objects generano Instance Objects che eseguono Attempt Objects.	Stringa

DynamoDB TableExists

Una preconditione per verificare l'esistenza della tabella DynamoDB.

Sintassi

Campi obbligatori	Description	Tipo di slot
role	Specifica il ruolo da utilizzare per eseguire la preconditione.	Stringa
tableName	Tabella DynamoDB da verificare.	Stringa

Campi opzionali	Description	Tipo di slot
attemptStatus	Lo stato segnalato più di recente dall'attività remota.	Stringa
attemptTimeout	Timeout per il completamento del lavoro in remoto. Se questo campo è impostato, un'attività remota che non viene completata entro il tempo impostato di avvio viene tentata di nuovo.	Periodo

Campi opzionali	Description	Tipo di slot
failureAndRerunModo	Descrive il comportamento del nodo consumer quando le dipendenze presentano un errore o vengono di nuovo eseguite	Enumerazione
lateAfterTimeout	Il tempo trascorso dopo l'inizio della pipeline entro il quale l'oggetto deve essere completato. Viene attivato solo quando il tipo di pianificazione non è impostato su. ondemand	Periodo
maximumRetries	Numero massimo di tentativi in caso di errore	Numero intero
onFail	Un'azione da eseguire quando l'oggetto corrente ha esito negativo.	Oggetto di riferimento, ad esempio «onFail»: {"ref":» myActionId «}
onLateAction	Azioni che devono essere attivate se un oggetto non è stato ancora pianificato o non è ancora completo.	Oggetto di riferimento, ad esempio "onLateAction«: {"ref":» myActionId «}
onSuccess	Un'operazione da eseguire quando l'oggetto corrente ha esito positivo.	Oggetto di riferimento, ad esempio «onSuccess»: {"ref":» myActionId «}
parent	Padre dell'oggetto corrente da cui saranno ereditati gli slot.	Oggetto di riferimento, ad esempio «parent»: {"ref":» myBaseObject Id "}
preconditionTimeout	Il periodo dall'inizio dopo il quale la condizione viene contrassegnata come non riuscita se ancora non è stata soddisfatta.	Periodo

Campi opzionali	Description	Tipo di slot
reportProgressTimeout	Timeout per chiamate successive di attività in remoto a reportProgress. Se impostato, le attività in remoto che non presentano avanzamenti nel periodo specificato potrebbero essere considerate bloccate e sono quindi oggetto di un altro tentativo.	Periodo
retryDelay	La durata del timeout tra due tentativi.	Periodo

Campi Runtime	Description	Tipo di slot
@activeInstances	Elenco di oggetti di istanze attive attualmente programmate.	Oggetto di riferimento, ad esempio «activeInstances»: {"ref":» myRunnableObject Id "}
@actualEndTime	L'ora in cui è terminata l'esecuzione di questo oggetto.	DateTime
@actualStartTime	L'ora in cui è stata avviata l'esecuzione di questo oggetto.	DateTime
cancellationReason	CancellationReason se questo oggetto è stato annullato.	Stringa
@cascadeFailedOn	Descrizione della catena di dipendenza che ha generato l'errore dell'oggetto.	Oggetto di riferimento, ad esempio "cascadeFailedOn«: {" ref":» myRunnableObject Id "}
currentRetryCount	Numero di volte in cui la precondizione è stato provata in questo tentativo.	Stringa

Campi Runtime	Description	Tipo di slot
emrStepLog	Log della fase EMR disponibili solo sui tentativi delle attività EMR	Stringa
errorId	ErrorId se l'oggetto non è riuscito.	Stringa
errorMessage	ErrorMessage se l'oggetto non è riuscito.	Stringa
errorStackTrace	Traccia dello stack di errore se l'oggetto non è riuscito.	Stringa
hadoopJobLog	Log delle attività Hadoop disponibili per le attività basate su EMR.	Stringa
hostname	Il nome host del client che si è aggiudicato il tentativo dell'attività.	Stringa
lastRetryTime	L'ultima volta in cui la condizione è stato provata all'interno di questo tentativo.	Stringa
nodo	Il nodo per il quale viene eseguita questa condizione	Oggetto di riferimento, ad esempio «node»: {"ref":» myRunnableObject Id "}
reportProgressTime	Il periodo di tempo più recente in cui l'attività remota ha segnalato un progresso.	DateTime
@scheduledEndTime	L'orario di termine della pianificazione per un oggetto	DateTime
@scheduledStartTime	L'orario di inizio della pianificazione per l'oggetto	DateTime
@status	Lo stato di questo oggetto.	Stringa
@version	Versione della pipeline con cui l'oggetto è stato creato.	Stringa

Campi Runtime	Description	Tipo di slot
@waitingOn	Descrizione dell'elenco di dipendenze per cui questo oggetto è in attesa.	Oggetto di riferimento, ad esempio «waitingOn»: {"ref":» myRunnableObject Id "}

Campi di sistema	Description	Tipo di slot
@error	Errore che descrive il formato oggetto errato	Stringa
@pipelineId	L'id della pipeline a cui appartiene questo oggetto	Stringa
@sphere	La sfera di un oggetto indica la propria posizione nel ciclo di vita: i Component Objects generano Instance Objects che eseguono Attempt Objects	Stringa

Exists

Verifica se esiste un oggetto nodo di dati.

Note

È consigliabile utilizzare precondizioni gestite dal sistema. Per ulteriori informazioni, consulta [Precondizioni](#).

Esempio

Di seguito è illustrato un esempio di questo tipo di oggetto. L'oggetto `InputData` fa riferimento a questo oggetto, `Ready`, più a un altro oggetto definito nello stesso file di definizione della pipeline. `CopyPeriod` è un oggetto `Schedule`.

```
{
  "id" : "InputData",
  "type" : "S3DataNode",
  "schedule" : { "ref" : "CopyPeriod" },
  "filePath" : "s3://example-bucket/InputData/#{@scheduledStartTime.format('YYYY-MM-dd-hh:mm')}.csv",
  "precondition" : { "ref" : "Ready" }
},
{
  "id" : "Ready",
  "type" : "Exists"
}
```

Sintassi

Campi opzionali	Description	Tipo di slot
attemptStatus	Lo stato segnalato più di recente dall'attività remota.	Stringa
attemptTimeout	Timeout per il completamento del lavoro in remoto. Se questo campo è impostato, un'attività remota che non viene completata entro il tempo impostato di avvio viene tentata di nuovo.	Periodo
failureAndRerunModo	Descrive il comportamento del nodo consumer quando le dipendenze presentano un errore o vengono di nuovo eseguite.	Enumerazione
lateAfterTimeout	Il tempo trascorso dopo l'inizio della pipeline entro il quale l'oggetto deve essere completato. Viene attivato solo quando il tipo di pianificazione non è impostato su. ondemand	Periodo
maximumRetries	Numero massimo di tentativi in caso di errore	Numero intero

Campi opzionali	Description	Tipo di slot
onFail	Un'azione da eseguire quando l'oggetto corrente ha esito negativo.	Oggetto di riferimento, ad esempio «onFail»: {"ref":» myActionId «}
onLateAction	Azioni che devono essere attivate se un oggetto non è stato ancora pianificato o non è ancora completo.	Oggetto di riferimento, ad esempio "onLateAction«: {"ref":» myActionId «}
onSuccess	Un'operazione da eseguire quando l'oggetto corrente ha esito positivo.	Oggetto di riferimento, ad esempio «onSuccess»: {"ref":» myActionId «}
parent	Padre dell'oggetto corrente da cui saranno ereditati gli slot.	Oggetto di riferimento, ad esempio «parent»: {"ref":» myBaseObject Id "}
preconditionTimeout	Il periodo dall'inizio dopo il quale la condizione viene contrassegnata come non riuscita se ancora non è stata soddisfatta.	Periodo
reportProgressTimeout	Timeout per chiamate successive di attività in remoto a reportProgress. Se impostato, le attività in remoto che non presentano avanzamenti nel periodo specificato potrebbero essere considerate bloccate e sono quindi oggetto di un altro tentativo.	Periodo
retryDelay	La durata del timeout tra due tentativi.	Periodo

Campi Runtime	Description	Tipo di slot
@activeInstances	Elenco di oggetti di istanze attive attualmente programmate.	Oggetto di riferimento, ad esempio «activeInstances»: {"ref":» myRunnableObject Id "}
@actualEndTime	L'ora in cui è terminata l'esecuzione di questo oggetto.	DateTime
@actualStartTime	L'ora in cui è stata avviata l'esecuzione di questo oggetto.	DateTime
cancellationReason	CancellationReason se questo oggetto è stato annullato.	Stringa
@cascadeFailedOn	Descrizione della catena di dipendenza che ha generato l'errore dell'oggetto.	Oggetto di riferimento, ad esempio "cascadeFailedOn«: {" ref":» myRunnableObject Id "}
emrStepLog	Log della fase EMR disponibili solo sui tentativi delle attività EMR	Stringa
errorId	ErrorId se l'oggetto non è riuscito.	Stringa
errorMessage	ErrorMessage se l'oggetto non è riuscito.	Stringa
errorStackTrace	Traccia dello stack di errore se l'oggetto non è riuscito.	Stringa
hadoopJobLog	Log delle attività Hadoop disponibili per le attività basate su EMR.	Stringa
hostname	Il nome host del client che si è aggiudicato il tentativo dell'attività.	Stringa

Campi Runtime	Description	Tipo di slot
nodo	Il nodo per il quale viene eseguita questa condizione.	Oggetto di riferimento, ad esempio «node»: {"ref":» myRunnableObject Id "}
reportProgressTime	Il periodo di tempo più recente in cui l'attività remota ha segnalato un progresso.	DateTime
@scheduledEndTime	L'orario di termine della pianificazione per un oggetto.	DateTime
@scheduledStartTime	L'orario di inizio della pianificazione per l'oggetto.	DateTime
@status	Lo stato di questo oggetto.	Stringa
@version	Versione della pipeline con cui l'oggetto è stato creato.	Stringa
@waitingOn	Descrizione dell'elenco di dipendenze per cui questo oggetto è in attesa.	Oggetto di riferimento, ad esempio «waitingOn»: {"ref":» myRunnableObject Id "}

Campi di sistema	Description	Tipo di slot
@error	Errore che descrive il formato oggetto errato.	Stringa
@pipelineId	L'id della pipeline a cui appartiene questo oggetto.	Stringa
@sphere	La sfera di un oggetto indica la propria posizione nel ciclo di vita: i Component Objects	Stringa

Campi di sistema	Description	Tipo di slot
	generano Instance Objects che eseguono Attempt Objects.	

Vedi anche

- [ShellCommandPrecondition](#)

S3 KeyExists

Verifica se esiste una chiave in un nodo di dati Amazon S3.

Esempio

Di seguito è illustrato un esempio di questo tipo di oggetto. La preconditione sarà attivata quando esiste la chiave `s3://mybucket/mykey`, a cui si fa riferimento tramite il parametro `s3Key`.

```
{
  "id" : "InputReady",
  "type" : "S3KeyExists",
  "role" : "test-role",
  "s3Key" : "s3://mybucket/mykey"
}
```

È inoltre possibile utilizzare `S3KeyExists` come preconditione nella seconda pipeline che attende il termine della prima pipeline. A tale scopo:

1. Scrivi un file su Amazon S3 al termine del completamento della prima pipeline.
2. Crea una preconditione `S3KeyExists` nella seconda pipeline.

Sintassi

Campi obbligatori	Description	Tipo di slot
role	Specifica il ruolo da utilizzare per eseguire la preconditione.	Stringa

Campi obbligatori	Description	Tipo di slot
s3Key	La chiave Amazon S3.	Stringa
Campi opzionali	Description	Tipo di slot
attemptStatus	Lo stato segnalato più di recente dall'attività remota.	Stringa
attemptTimeout	Timeout prima di tentare nuovamente di completare il lavoro in remoto. Se impostato , un'attività remota che non viene completat a entro il tempo impostato dopo l'avvio viene tentata di nuovo.	Periodo
failureAndRerunModa	Descrive il comportamento del nodo consumer quando le dipendenze presentano un errore o vengono di nuovo eseguite.	Enumerazione
lateAfterTimeout	Il tempo trascorso dopo l'inizio della pipeline entro il quale l'oggetto deve essere completat o. Viene attivato solo quando il tipo di pianifica zione non è impostato su. ondemand	Periodo
maximumRetries	Numero massimo di tentativi che vengono avviati in caso di errore.	Numero intero
onFail	Un'azione da eseguire quando l'oggetto corrente ha esito negativo.	Oggetto di riferimento, ad esempio «onFail»: {"ref":» myActionId «}
onLateAction	Azioni che devono essere attivate se un oggetto non è stato ancora pianificato o non è ancora completo.	Oggetto di riferimen to, ad esempio "onLateAction«: {" ref":» myActionId «}

Campi opzionali	Description	Tipo di slot
onSuccess	Un'operazione da eseguire quando l'oggetto corrente ha esito positivo.	Oggetto di riferimento, ad esempio «onSuccess»: {"ref":» myActionId «}
parent	Padre dell'oggetto corrente da cui saranno ereditati gli slot.	Oggetto di riferimento, ad esempio «parent»: {"ref":» myBaseObject Id "}
preconditionTimeout	Il periodo dall'inizio dopo il quale la condizione viene contrassegnata come non riuscita se ancora non è stata soddisfatta.	Periodo
reportProgressTimeout	Timeout per chiamate successive di attività in remoto a <code>reportProgress</code> . Se impostato, le attività in remoto che non presentano avanzamenti nel periodo specificato potrebbero essere considerate bloccate e sono quindi oggetto di un altro tentativo.	Periodo
retryDelay	La durata del timeout tra due tentativi consecutivi.	Periodo

Campi Runtime	Description	Tipo di slot
@activeInstances	Elenco di oggetti di istanze attive attualmente programmate.	Oggetto di riferimento, ad esempio «activeInstances»: {"ref":» myRunnableObject Id "}
@actualEndTime	L'ora in cui è terminata l'esecuzione di questo oggetto.	DateTime

Campi Runtime	Description	Tipo di slot
@actualStartTime	L'ora in cui è stata avviata l'esecuzione di questo oggetto.	DateTime
cancellationReason	CancellationReason se questo oggetto è stato annullato.	Stringa
@cascadeFailedOn	Descrizione della catena di dipendenza che ha generato l'errore dell'oggetto.	Oggetto di riferimento, ad esempio "cascadeFailedOn«: {" ref»:» myRunnableObject Id "}
currentRetryCount	Numero di volte in cui la precondizione è stato provata in questo tentativo.	Stringa
emrStepLog	Log della fase EMR disponibili solo sui tentativi delle attività EMR	Stringa
errorId	ErrorId se l'oggetto non è riuscito.	Stringa
errorMessage	ErrorMessage se l'oggetto non è riuscito.	Stringa
errorStackTrace	Traccia dello stack di errore se l'oggetto non è riuscito.	Stringa
hadoopJobLog	Log delle attività Hadoop disponibili per le attività basate su EMR.	Stringa
hostname	Il nome host del client che si è aggiudicato il tentativo dell'attività.	Stringa
lastRetryTime	L'ultima volta in cui la precondizione è stato provata all'interno di questo tentativo.	Stringa

Campi Runtime	Description	Tipo di slot
nodo	Il nodo per il quale viene eseguita questa condizione	Oggetto di riferimento, ad esempio «node»: {"ref":» myRunnableObject Id "}
reportProgressTime	Il periodo di tempo più recente in cui l'attività remota ha segnalato un progresso.	DateTime
@scheduledEndTime	L'orario di termine della pianificazione per un oggetto	DateTime
@scheduledStartTime	L'orario di inizio della pianificazione per l'oggetto	DateTime
@status	Lo stato di questo oggetto.	Stringa
@version	Versione della pipeline con cui l'oggetto è stato creato.	Stringa
@waitingOn	Descrizione dell'elenco di dipendenze per cui questo oggetto è in attesa.	Oggetto di riferimento, ad esempio «waitingOn»: {"ref":» myRunnableObject Id "}

Campi di sistema	Description	Tipo di slot
@error	Errore che descrive il formato oggetto errato	Stringa
@pipelineId	L'id della pipeline a cui appartiene questo oggetto	Stringa
@sphere	La sfera di un oggetto indica la propria posizione nel ciclo di vita: i Component Objects	Stringa

Campi di sistema	Description	Tipo di slot
	generano Instance Objects che eseguono Attempt Objects	

Vedi anche

- [ShellCommandPrecondition](#)

S3 PrefixNotEmpty

Una preconditione per verificare la presenza degli oggetti Amazon S3 con il prefisso specificato (rappresentato come URI).

Esempio

Di seguito è riportato un esempio di questo tipo di oggetto utilizzando campi obbligatori, facoltativi e di espressioni.

```
{
  "id" : "InputReady",
  "type" : "S3PrefixNotEmpty",
  "role" : "test-role",
  "s3Prefix" : "#{node.filePath}"
}
```

Sintassi

Campi obbligatori	Description	Tipo di slot
role	Specifica il ruolo da utilizzare per eseguire la preconditione.	Stringa
s3Prefix	Il prefisso Amazon S3 per verificare l'esistenza di oggetti.	Stringa

Campi opzionali	Description	Tipo di slot
attemptStatus	Lo stato segnalato più di recente dall'attività remota.	Stringa
attemptTimeout	Timeout per il completamento del lavoro in remoto. Se questo campo è impostato, un'attività remota che non viene completata entro il tempo impostato di avvio viene tentata di nuovo.	Periodo
failureAndRerunModo	Descrive il comportamento del nodo consumer quando le dipendenze presentano un errore o vengono di nuovo eseguite	Enumerazione
lateAfterTimeout	Il tempo trascorso dopo l'inizio della pipeline entro il quale l'oggetto deve essere completato. Viene attivato solo quando il tipo di pianificazione non è impostato su. ondemand	Periodo
maximumRetries	Numero massimo di tentativi in caso di errore	Numero intero
onFail	Un'azione da eseguire quando l'oggetto corrente ha esito negativo.	Oggetto di riferimento, ad esempio «onFail»: {"ref":» myActionId «}
onLateAction	Azioni che devono essere attivate se un oggetto non è stato ancora pianificato o non è ancora completo.	Oggetto di riferimento, ad esempio "onLateAction«: {"ref":» myActionId «}
onSuccess	Un'operazione da eseguire quando l'oggetto corrente ha esito positivo.	Oggetto di riferimento, ad esempio «onSuccess»: {"ref":» myActionId «}
parent	Padre dell'oggetto corrente da cui saranno ereditati gli slot.	Oggetto di riferimento, ad esempio «parent»:

Campi opzionali	Description	Tipo di slot
		<code>{"ref":» myBaseObject Id "}</code>
<code>preconditionTimeout</code>	Il periodo dall'inizio dopo il quale la condizione viene contrassegnata come non riuscita se ancora non è stata soddisfatta.	Periodo
<code>reportProgressTimeout</code>	Timeout per chiamate successive di attività in remoto a <code>reportProgress</code> . Se impostato, le attività in remoto che non presentano avanzamenti nel periodo specificato potrebbero essere considerate bloccate e sono quindi oggetto di un altro tentativo.	Periodo
<code>retryDelay</code>	La durata del timeout tra due tentativi.	Periodo

Campi Runtime	Description	Tipo di slot
<code>@activeInstances</code>	Elenco di oggetti di istanze attive attualmente programmate.	Oggetto di riferimento, ad esempio « <code>activeInstances</code> »: <code>{"ref":» myRunnableObject Id "}</code>
<code>@actualEndTime</code>	L'ora in cui è terminata l'esecuzione di questo oggetto.	DateTime
<code>@actualStartTime</code>	L'ora in cui è stata avviata l'esecuzione di questo oggetto.	DateTime
<code>cancellationReason</code>	<code>CancellationReason</code> se questo oggetto è stato annullato.	Stringa
<code>@cascadeFailedOn</code>	Descrizione della catena di dipendenza che ha generato l'errore dell'oggetto.	Oggetto di riferimento, ad esempio

Campi Runtime	Description	Tipo di slot
		"cascadeFailedOn«: {"ref»:» myRunnabl eObject Id "
currentRetryCount	Numero di volte in cui la precondizione è stato provata in questo tentativo.	Stringa
emrStepLog	Log della fase EMR disponibili solo sui tentativi delle attività EMR	Stringa
errorId	ErrorId se l'oggetto non è riuscito.	Stringa
errorMessage	ErrorMessage se l'oggetto non è riuscito.	Stringa
errorStackTrace	Traccia dello stack di errore se l'oggetto non è riuscito.	Stringa
hadoopJobLog	Log delle attività Hadoop disponibili per le attività basate su EMR.	Stringa
hostname	Il nome host del client che si è aggiudicato il tentativo dell'attività.	Stringa
lastRetryTime	L'ultima volta in cui la precondizione è stato provata all'interno di questo tentativo.	Stringa
nodo	Il nodo per il quale viene eseguita questa precondizione.	Oggetto di riferimento, ad esempio «node»: {"ref»:» myRunnabl eObject Id "}
reportProgressTime	Il periodo di tempo più recente in cui l'attività remota ha segnalato un progresso.	DateTime
@scheduledEndTime	L'orario di termine della pianificazione per un oggetto.	DateTime

Campi Runtime	Description	Tipo di slot
@scheduledStartTime	L'orario di inizio della pianificazione per l'oggetto.	DateTime
@status	Lo stato di questo oggetto.	Stringa
@version	Versione della pipeline con cui l'oggetto è stato creato.	Stringa
@waitingOn	Descrizione dell'elenco di dipendenze per cui questo oggetto è in attesa.	Oggetto di riferimento, ad esempio «waitingOn»: {"ref":» myRunnableObject Id "}

Campi di sistema	Description	Tipo di slot
@error	Errore che descrive il formato oggetto errato	Stringa
@pipelineId	L'id della pipeline a cui appartiene questo oggetto	Stringa
@sphere	La sfera di un oggetto indica la propria posizione nel ciclo di vita: i Component Objects generano Instance Objects che eseguono Attempt Objects	Stringa

Vedi anche

- [ShellCommandPrecondition](#)

ShellCommandPrecondition

Un comando shell Linux/Unix che può essere eseguito come preconditione.

Esempio

Di seguito è illustrato un esempio di questo tipo di oggetto.

```
{
  "id" : "VerifyDataReadiness",
  "type" : "ShellCommandPrecondition",
  "command" : "perl check-data-ready.pl"
}
```

Sintassi

Gruppo richiesto (uno dei seguenti è obbligatorio)	Description	Tipo di slot
command	Il comando da eseguire. Questo valore ed eventuali parametri associati devono funzionare nell'ambiente da cui si sta eseguendo il Task Runner.	Stringa
scriptUri	Un percorso URI di Amazon S3 per un file da scaricare ed eseguire come comando shell. Deve essere presente solo uno scriptUri o un campo di comando. Se lo scriptUri non è in grado di utilizzare i parametri, utilizzare il comando.	Stringa

Campi opzionali	Description	Tipo di slot
attemptStatus	Lo stato segnalato più di recente dall'attività remota.	Stringa
attemptTimeout	Timeout per il completamento del lavoro in remoto. Se questo campo è impostato, un'attività remota che non viene completata entro	Periodo

Campi opzionali	Description	Tipo di slot
	il tempo impostato di avvio viene tentata di nuovo.	
failureAndRerunModo	Descrive il comportamento del nodo consumer quando le dipendenze presentano un errore o vengono di nuovo eseguite	Enumerazione
lateAfterTimeout	Il tempo trascorso dopo l'inizio della pipeline entro il quale l'oggetto deve essere completato. Viene attivato solo quando il tipo di pianificazione non è impostato su. ondemand	Periodo
maximumRetries	Numero massimo di tentativi in caso di errore	Numero intero
onFail	Un'azione da eseguire quando l'oggetto corrente ha esito negativo.	Oggetto di riferimento, ad esempio «onFail»: {"ref":» myActionId «}
onLateAction	Azioni che devono essere attivate se un oggetto non è stato ancora pianificato o non è ancora completo.	Oggetto di riferimento, ad esempio "onLateAction«: {"ref":» myActionId «}
onSuccess	Un'operazione da eseguire quando l'oggetto corrente ha esito positivo.	Oggetto di riferimento, ad esempio «onSuccess»: {"ref":» myActionId «}
parent	Padre dell'oggetto corrente da cui saranno ereditati gli slot.	Oggetto di riferimento, ad esempio «parent»: {"ref":» myBaseObject Id "}
preconditionTimeout	Il periodo dall'inizio dopo il quale la condizione viene contrassegnata come non riuscita se ancora non è stata soddisfatta.	Periodo

Campi opzionali	Description	Tipo di slot
reportProgressTimeout	Timeout per chiamate successive di attività in remoto a reportProgress. Se impostato , le attività in remoto che non presentano avanzamenti nel periodo specificato potrebbero essere considerate bloccate e sono quindi oggetto di un altro tentativo.	Periodo
retryDelay	La durata del timeout tra due tentativi.	Periodo
scriptArgument	Argomento da passare allo script della shell	Stringa
stderr	Il percorso Amazon S3 che riceve i messaggi di errore di sistema reindirizzati dal comando. Se utilizzi il runsOn campo, deve trattarsi di un percorso Amazon S3 a causa della natura transitoria della risorsa che esegue la tua attività. Tuttavia, se specifichi il campo workerGroup , viene autorizzato un percorso file locale.	Stringa
stdout	Il percorso Amazon S3 che riceve l'output reindirizzato dal comando. Se utilizzi il runsOn campo, deve trattarsi di un percorso Amazon S3 a causa della natura transitoria della risorsa che esegue la tua attività. Tuttavia, se specifichi il campo workerGroup , viene autorizzato un percorso file locale.	Stringa

Campi Runtime	Description	Tipo di slot
@activeInstances	Elenco di oggetti di istanze attive attualmente programmate.	Oggetto di riferimento, ad esempio «activeInstances»: {"ref":» Id "myRunnableObject

Campi Runtime	Description	Tipo di slot
@actualEndTime	L'ora in cui è terminata l'esecuzione di questo oggetto.	DateTime
@actualStartTime	L'ora in cui è stata avviata l'esecuzione di questo oggetto.	DateTime
cancellationReason	CancellationReason se questo oggetto è stato annullato.	Stringa
@cascadeFailedOn	Descrizione della catena di dipendenza che ha generato l'errore dell'oggetto.	Oggetto di riferimento, ad esempio "cascadeFailedOn«: {"ref»:» myRunnableObject Id "
emrStepLog	Log della fase EMR disponibili solo sui tentativi delle attività EMR	Stringa
errorId	ErrorId se l'oggetto non è riuscito.	Stringa
errorMessage	ErrorMessage se l'oggetto non è riuscito.	Stringa
errorStackTrace	Traccia dello stack di errore se l'oggetto non è riuscito.	Stringa
hadoopJobLog	Log delle attività Hadoop disponibili per le attività basate su EMR.	Stringa
hostname	Il nome host del client che si è aggiudicato il tentativo dell'attività.	Stringa
nodo	Il nodo per il quale viene eseguita questa precondizione	Oggetto di riferimento, ad esempio «node»: {"ref»:» myRunnableObject Id "

Campi Runtime	Description	Tipo di slot
reportProgressTime	Il periodo di tempo più recente in cui l'attività remota ha segnalato un progresso.	DateTime
@scheduledEndTime	L'orario di termine della pianificazione per un oggetto	DateTime
@scheduledStartTime	L'orario di inizio della pianificazione per l'oggetto	DateTime
@status	Lo stato di questo oggetto.	Stringa
@version	Versione della pipeline con cui l'oggetto è stato creato.	Stringa
@waitingOn	Descrizione dell'elenco di dipendenze per cui questo oggetto è in attesa.	Oggetto di riferimento, ad esempio «waitingOn»: {"ref":» myRunnableObject Id "}

Campi di sistema	Description	Tipo di slot
@error	Errore che descrive il formato oggetto errato	Stringa
@pipelineId	L'id della pipeline a cui appartiene questo oggetto	Stringa
@sphere	La sfera di un oggetto indica la propria posizione nel ciclo di vita: i Component Objects generano Instance Objects che eseguono Attempt Objects	Stringa

Vedi anche

- [ShellCommandActivity](#)

- [Exists](#)

Database

Di seguito sono elencati gli oggetti di database AWS Data Pipeline:

Oggetti

- [JdbcDatabase](#)
- [RdsDatabase](#)
- [RedshiftDatabase](#)

JdbcDatabase

Rimuove un database JDBC.

Esempio

Di seguito è illustrato un esempio di questo tipo di oggetto.

```
{
  "id" : "MyJdbcDatabase",
  "type" : "JdbcDatabase",
  "connectionString" : "jdbc:redshift://hostname:portnumber/dbname",
  "jdbcDriverClass" : "com.amazon.redshift.jdbc41.Driver",
  "jdbcDriverJarUri" : "s3://redshift-downloads/drivers/RedshiftJDBC41-1.1.6.1006.jar",
  "username" : "user_name",
  "*password" : "my_password"
}
```

Sintassi

Campi obbligatori	Description	Tipo di slot
connectionString	La stringa di connessione JDBC per accedere al database.	Stringa
jdbcDriverClass	La classe del driver da caricare prima di stabilire la connessione JDBC.	Stringa

Campi obbligatori	Description	Tipo di slot
*password	La password da fornire.	Stringa
username	Il nome utente da fornire durante la connessione al database.	Stringa

Campi opzionali	Description	Tipo di slot
databaseName	Nome del database logico a cui collegarsi	Stringa
jdbcDriverJarUri	Il percorso in Amazon S3 del file JAR del driver JDBC utilizzato per la connessione al database. AWS Data Pipeline deve avere l'autorizzazione a leggere questo file JAR.	Stringa
jdbcProperties	Coppie della forma A=B da impostare come proprietà sulle connessioni JDBC per questo database.	Stringa
parent	Padre dell'oggetto corrente da cui saranno ereditati gli slot.	Oggetto di riferimento, ad esempio «parent»: {"ref":» myBaseObject Id "}

Campi Runtime	Description	Tipo di slot
@version	Versione della pipeline con cui l'oggetto è stato creato.	Stringa

Campi di sistema	Description	Tipo di slot
@error	Errore che descrive il formato oggetto errato.	Stringa

Campi di sistema	Description	Tipo di slot
@pipelineId	L'ID della pipeline a cui appartiene questo oggetto.	Stringa
@sphere	La sfera di un oggetto indica la propria posizione nel ciclo di vita: i Component Objects generano Instance Objects che eseguono Attempt Objects.	Stringa

RdsDatabase

Definisce un database Amazon RDS.

Note

RdsDatabase non supporta Aurora. Usalo [the section called "JdbcDatabase"](#) per Aurora, invece.

Esempio

Di seguito è illustrato un esempio di questo tipo di oggetto.

```
{
  "id" : "MyRdsDatabase",
  "type" : "RdsDatabase",
  "region" : "us-east-1",
  "username" : "user_name",
  "*password" : "my_password",
  "rdsInstanceId" : "my_db_instance_identifier"
}
```

Per il motore Oracle, il campo `jdbcDriverJarUri` è obbligatorio ed è possibile specificare il seguente driver: <http://www.oracle.com/technetwork/database/features/jdbc/jdbc-drivers-12c-download-1958347.html>. Per il motore SQL, il campo `jdbcDriverJarUri` è obbligatorio ed è possibile specificare il seguente driver: <https://www.microsoft.com/en-us/>

download/details.aspx?displaylang=en&id=11774. Per i motori MySQL e PostgreSQL, il campo `jdbcDriverJarUri` è facoltativo.

Sintassi

Campi obbligatori	Description	Tipo di slot
<code>*password</code>	La password da fornire.	Stringa
<code>rdsInstanceCld</code>	La <code>DBInstanceIdentifier</code> proprietà dell'istanza DB.	Stringa
<code>username</code>	Il nome utente da fornire durante la connessione al database.	Stringa

Campi opzionali	Description	Tipo di slot
<code>databaseName</code>	Nome del database logico a cui collegarsi	Stringa
<code>jdbcDriverJarUri</code>	Il percorso in Amazon S3 del file JAR del driver JDBC utilizzato per la connessione al database. AWS Data Pipeline deve avere l'autorizzazione a leggere questo file JAR. Per i motori MySQL e PostgreSQL, il driver predefinito viene utilizzato se questo campo non è specificato, ma è possibile sostituire l'impostazione predefinita utilizzando tale campo. Per i motori Oracle e SQL Server, questo campo è obbligatorio.	Stringa
<code>jdbcProperties</code>	Coppie della forma <code>A=B</code> da impostare come proprietà sulle connessioni JDBC per questo database.	Stringa
<code>parent</code>	Padre dell'oggetto corrente da cui saranno ereditati gli slot.	Oggetto di riferimento, ad esempio «parent»:

Campi opzionali	Description	Tipo di slot
		{"ref»:» myBaseObject Id "}
Regione	Il codice per la regione in cui esiste il database. Ad esempio, us-east-1.	Stringa

Campi Runtime	Description	Tipo di slot
@version	Versione della pipeline con cui l'oggetto è stato creato.	Stringa

Campi di sistema	Description	Tipo di slot
@error	Errore che descrive il formato oggetto errato.	Stringa
@pipelineId	L'ID della pipeline a cui appartiene questo oggetto.	Stringa
@sphere	La sfera di un oggetto indica la propria posizione nel ciclo di vita: i Component Objects generano Instance Objects che eseguono Attempt Objects.	Stringa

RedshiftDatabase

Definisce un database Amazon Redshift. `RedshiftDatabase` rappresenta le proprietà del database utilizzato dalla pipeline.

Esempio

Di seguito è illustrato un esempio di questo tipo di oggetto.

```
{
```

```

"id" : "MyRedshiftDatabase",
"type" : "RedshiftDatabase",
"clusterId" : "myRedshiftClusterId",
"username" : "user_name",
"*password" : "my_password",
"databaseName" : "database_name"
}

```

Per impostazione predefinita, l'oggetto usa il driver Postgres, che richiede il campo `clusterId`. Per utilizzare il driver Amazon Redshift, specifica invece la stringa di connessione al database Amazon Redshift dalla console Amazon Redshift (inizia con «jdbc:redshift:») nel campo. `connectionString`

Sintassi

Campi obbligatori	Description	Tipo di slot
*password	La password da fornire.	Stringa
username	Il nome utente da fornire durante la connessione al database.	Stringa

Gruppo richiesto (uno dei seguenti è obbligatorio)	Description	Tipo di slot
clusterId	L'identificatore fornito dall'utente al momento della creazione del cluster Amazon Redshift. Ad esempio, se l'endpoint per il tuo cluster Amazon Redshift è <code>mydb.example.us-east-1.redshift.amazonaws.com</code> , l'identificatore corretto è <code>mydb</code> . Nella console Amazon Redshift, puoi ottenere questo valore da Cluster Identifier o Cluster Name.	Stringa
connectionString	L'endpoint JDBC per la connessione a un'istanza a Amazon Redshift di proprietà di un account	Stringa

Gruppo richiesto (uno dei seguenti è obbligatorio)	Description	Tipo di slot
	diverso dalla pipeline. Non è possibile specificare <code>connectionString</code> e <code>clusterId</code> .	

Campi opzionali	Description	Tipo di slot
<code>databaseName</code>	Nome del database logico a cui collegarsi.	Stringa
<code>jdbcProperties</code>	Coppie della forma A=B da impostare come proprietà sulle connessioni JDBC per questo database.	Stringa
<code>parent</code>	Padre dell'oggetto corrente da cui vengono ereditati gli slot.	Oggetto di riferimento, ad esempio «parent»: {"ref":» Id "} myBaseObject
Regione	Il codice per la regione in cui esiste il database. Ad esempio, us-east-1.	Enumerazione

Campi Runtime	Description	Tipo di slot
<code>@version</code>	Versione della pipeline con cui l'oggetto è stato creato.	Stringa

Campi di sistema	Description	Tipo di slot
<code>@error</code>	Errore che descrive il formato oggetto errato.	Stringa

Campi di sistema	Description	Tipo di slot
@pipelineId	L'ID della pipeline a cui appartiene questo oggetto.	Stringa
@sphere	La sfera di un oggetto indica la propria posizione nel ciclo di vita: i Component Objects generano Instance Objects che eseguono Attempt Objects.	Stringa

Formati dei dati

Di seguito sono elencati gli oggetti dei formati di dati AWS Data Pipeline:

Oggetti

- [Formato dei dati CSV](#)
- [Formato di dati personalizzato](#)
- [DynamoDB DataFormat](#)
- [DynamoDB ExportDataFormat](#)
- [RegEx Formato dei dati](#)
- [Formato dei dati TSV](#)

Formato dei dati CSV

Un formato di dati separati da virgola in cui il separatore di colonna è una virgola e il separatore di record è un carattere in una nuova riga.

Esempio

Di seguito è illustrato un esempio di questo tipo di oggetto.

```
{
  "id" : "MyOutputDataType",
  "type" : "CSV",
  "column" : [
    "Name STRING",
```

```

    "Score INT",
    "DateOfBirth TIMESTAMP"
  ]
}
```

Sintassi

Campi opzionali	Description	Tipo di slot
column	Il nome della colonna con il tipo di dati specifica to da ogni campo per i dati descritti da questo nodo di dati. Ex: hostname STRING Per più valori, utilizzare i nomi delle colonne e i tipi di dati separati da uno spazio.	Stringa
escapeChar	Carattere, ad esempio "\", che indica al parser di ignorare il carattere successivo.	Stringa
parent	Padre dell'oggetto corrente da cui saranno ereditati gli slot.	Oggetto di riferimento, ad esempio «parent»: {"ref":» myBaseObject Id "}

Campi Runtime	Description	Tipo di slot
@version	Versione della pipeline con cui l'oggetto è stato creato.	Stringa

Campi di sistema	Description	Tipo di slot
@error	Errore che descrive il formato oggetto errato	Stringa
@pipelineId	L'id della pipeline a cui appartiene questo oggetto	Stringa

Campi di sistema	Description	Tipo di slot
@sphere	La sfera di un oggetto indica la propria posizione nel ciclo di vita: i Component Objects generano Instance Objects che eseguono Attempt Objects	Stringa

Formato di dati personalizzato

Un formato di dati personalizzato definito da una combinazione di un determinato separatore di colonne, da un separatore di record e da un carattere escape.

Esempio

Di seguito è illustrato un esempio di questo tipo di oggetto.

```
{
  "id" : "MyOutputDataType",
  "type" : "Custom",
  "columnSeparator" : ",",
  "recordSeparator" : "\n",
  "column" : [
    "Name STRING",
    "Score INT",
    "DateOfBirth TIMESTAMP"
  ]
}
```

Sintassi

Campi obbligatori	Description	Tipo di slot
columnSeparator	Carattere che indica la fine di una colonna in un file di dati.	Stringa

Campi opzionali	Description	Tipo di slot
column	Il nome della colonna con il tipo di dati specificato da ogni campo per i dati descritti da questo nodo di dati. Ex: hostname STRING Per più valori, utilizzare i nomi delle colonne e i tipi di dati separati da uno spazio.	Stringa
parent	Padre dell'oggetto corrente da cui saranno ereditati gli slot.	Oggetto di riferimento, ad esempio «parent»: {"ref":» myBaseObject Id "}
recordSeparator	Carattere che indica la fine di una riga in un file di dati, ad esempio "\n". Supporta solo caratteri singoli.	Stringa

Campi Runtime	Description	Tipo di slot
@version	Versione della pipeline con cui l'oggetto è stato creato.	Stringa

Campi di sistema	Description	Tipo di slot
@error	Errore che descrive il formato oggetto errato	Stringa
@pipelineId	L'id della pipeline a cui appartiene questo oggetto	Stringa
@sphere	La sfera di un oggetto indica la propria posizione nel ciclo di vita: i Component Objects generano Instance Objects che eseguono Attempt Objects	Stringa

DynamoDB DataFormat

Applica uno schema a una tabella DynamoDB per renderla accessibile tramite una query Hive. DynamoDBDataFormat viene utilizzato con un HiveActivity oggetto e un DynamoDBDataNode input e output. DynamoDBDataFormat richiede che tu specifichi tutte le colonne nella tua query Hive. Per una maggiore flessibilità nello specificare determinate colonne in una query Hive o nel supporto di Amazon S3, consulta. [DynamoDB ExportDataFormat](#)

Note

I tipi DynamoDB Boolean non sono mappati sui tipi Hive Boolean. Tuttavia, è possibile mappare valori interi DynamoDB pari a 0 o 1 su tipi Hive Boolean.

Esempio

L'esempio seguente mostra come usare DynamoDBDataFormat per assegnare uno schema a un input DynamoDBDataNode, che consente a un oggetto HiveActivity di accedere ai dati in base alle colonne denominate e copiare i dati in un output DynamoDBDataNode.

```
{
  "objects": [
    {
      "id" : "Exists.1",
      "name" : "Exists.1",
      "type" : "Exists"
    },
    {
      "id" : "DataFormat.1",
      "name" : "DataFormat.1",
      "type" : "DynamoDBDataFormat",
      "column" : [
        "hash STRING",
        "range STRING"
      ]
    },
    {
      "id" : "DynamoDBDataNode.1",
      "name" : "DynamoDBDataNode.1",
      "type" : "DynamoDBDataNode",
      "tableName" : "$INPUT_TABLE_NAME",
```

```

    "schedule" : { "ref" : "ResourcePeriod" },
    "dataFormat" : { "ref" : "DataFormat.1" }
  },
  {
    "id" : "DynamoDBDataNode.2",
    "name" : "DynamoDBDataNode.2",
    "type" : "DynamoDBDataNode",
    "tableName" : "$OUTPUT_TABLE_NAME",
    "schedule" : { "ref" : "ResourcePeriod" },
    "dataFormat" : { "ref" : "DataFormat.1" }
  },
  {
    "id" : "EmrCluster.1",
    "name" : "EmrCluster.1",
    "type" : "EmrCluster",
    "schedule" : { "ref" : "ResourcePeriod" },
    "masterInstanceType" : "m1.small",
    "keyPair" : "$KEYPAIR"
  },
  {
    "id" : "HiveActivity.1",
    "name" : "HiveActivity.1",
    "type" : "HiveActivity",
    "input" : { "ref" : "DynamoDBDataNode.1" },
    "output" : { "ref" : "DynamoDBDataNode.2" },
    "schedule" : { "ref" : "ResourcePeriod" },
    "runsOn" : { "ref" : "EmrCluster.1" },
    "hiveScript" : "insert overwrite table ${output1} select * from ${input1} ;"
  },
  {
    "id" : "ResourcePeriod",
    "name" : "ResourcePeriod",
    "type" : "Schedule",
    "period" : "1 day",
    "startDateTime" : "2012-05-04T00:00:00",
    "endDateTime" : "2012-05-05T00:00:00"
  }
]
}

```

Sintassi

Campi opzionali	Description	Tipo di slot
column	Il nome della colonna con il tipo di dati specificato da ogni campo per i dati descritti da questo nodo di dati. Ad esempio, <code>hostname STRING</code> . Per più valori, utilizzare i nomi delle colonne e i tipi di dati separati da uno spazio.	Stringa
parent	Padre dell'oggetto corrente da cui saranno ereditati gli slot.	Oggetto di riferimento, ad esempio «parent»: <code>{"ref":» myBaseObject Id "}</code>

Campi Runtime	Description	Tipo di slot
@version	La versione della pipeline utilizzata per creare l'oggetto.	Stringa

Campi di sistema	Description	Tipo di slot
@error	Errore che descrive l'oggetto con il formato errato.	Stringa
@pipelineId	L'id della pipeline a cui appartiene questo oggetto.	Stringa
@sphere	La sfera di un oggetto indica la propria posizione nel ciclo di vita: i Component Objects generano Instance Objects che eseguono Attempt Objects.	Stringa

DynamoDB ExportDataFormat

Applica uno schema a una tabella DynamoDB per renderla accessibile tramite una query Hive. Utilizzare `DynamoDBExportDataFormat` con un oggetto `HiveCopyActivity` e `DynamoDBDataNode` o con input e output `S3DataNode`. `DynamoDBExportDataFormat` ha i seguenti benefici:

- Fornisce supporto sia per DynamoDB che per Amazon S3
- Consente di filtrare i dati in base a determinate colonne nella query Hive
- Esporta tutti gli attributi da DynamoDB anche se hai uno schema sparso

Note

I tipi DynamoDB Boolean non sono mappati sui tipi Hive Boolean. Tuttavia, è possibile mappare valori interi DynamoDB pari a 0 o 1 su tipi Hive Boolean.

Esempio

L'esempio seguente mostra come usare `HiveCopyActivity` e `DynamoDBExportDataFormat` per copiare i dati da una versione `DynamoDBDataNode` a un'altra, mentre i dati vengono filtrati in base a un timestamp.

```
{
  "objects": [
    {
      "id" : "DataFormat.1",
      "name" : "DataFormat.1",
      "type" : "DynamoDBExportDataFormat",
      "column" : "timeStamp BIGINT"
    },
    {
      "id" : "DataFormat.2",
      "name" : "DataFormat.2",
      "type" : "DynamoDBExportDataFormat"
    },
    {
      "id" : "DynamoDBDataNode.1",
      "name" : "DynamoDBDataNode.1",
      "type" : "DynamoDBDataNode",
    }
  ]
}
```

```

    "tableName" : "item_mapped_table_restore_temp",
    "schedule" : { "ref" : "ResourcePeriod" },
    "dataFormat" : { "ref" : "DataFormat.1" }
  },
  {
    "id" : "DynamoDBDataNode.2",
    "name" : "DynamoDBDataNode.2",
    "type" : "DynamoDBDataNode",
    "tableName" : "restore_table",
    "region" : "us_west_1",
    "schedule" : { "ref" : "ResourcePeriod" },
    "dataFormat" : { "ref" : "DataFormat.2" }
  },
  {
    "id" : "EmrCluster.1",
    "name" : "EmrCluster.1",
    "type" : "EmrCluster",
    "schedule" : { "ref" : "ResourcePeriod" },
    "masterInstanceType" : "m1.xlarge",
    "coreInstanceCount" : "4"
  },
  {
    "id" : "HiveTransform.1",
    "name" : "Hive Copy Transform.1",
    "type" : "HiveCopyActivity",
    "input" : { "ref" : "DynamoDBDataNode.1" },
    "output" : { "ref" : "DynamoDBDataNode.2" },
    "schedule" : { "ref" : "ResourcePeriod" },
    "runsOn" : { "ref" : "EmrCluster.1" },
    "filterSql" : "`timeStamp` > unix_timestamp(\"#{@scheduledStartTime}\", \"yyyy-MM-dd'T'HH:mm:ss\")"
  },
  {
    "id" : "ResourcePeriod",
    "name" : "ResourcePeriod",
    "type" : "Schedule",
    "period" : "1 Hour",
    "startDateTime" : "2013-06-04T00:00:00",
    "endDateTime" : "2013-06-04T01:00:00"
  }
]
}

```

Sintassi

Campi opzionali	Description	Tipo di slot
column	Il nome della colonna con il tipo di dati specificato da ogni campo per i dati descritti da questo nodo di dati. Ex: hostname STRING	Stringa
parent	Padre dell'oggetto corrente da cui saranno ereditati gli slot.	Oggetto di riferimento, ad esempio «parent»: {"ref":» myBaseObject Id "}

Campi Runtime	Description	Tipo di slot
@version	Versione della pipeline con cui l'oggetto è stato creato.	Stringa

Campi di sistema	Description	Tipo di slot
@error	Errore che descrive il formato oggetto errato	Stringa
@pipelineId	L'id della pipeline a cui appartiene questo oggetto	Stringa
@sphere	La sfera di un oggetto indica la propria posizione nel ciclo di vita: i Component Objects generano Instance Objects che eseguono Attempt Objects	Stringa

RegEx Formato dei dati

Un formato di dati personalizzato definito da un'espressione regolare.

Esempio

Di seguito è illustrato un esempio di questo tipo di oggetto.

```
{
  "id" : "MyInputDataType",
  "type" : "Regex",
  "inputRegex" : "([ ]*) ([ ]*) ([ ]*) (-|\\[[^\\]]*\\]) ([^ \\"]*|\"[^\"]*\\") (-|[0-9]*) (-|[0-9]*)?(: ([^ \\"]*|\"[^\"]*\\") ([^ \\"]*|\"[^\"]*\\\"))?)?",
  "outputFormat" : "%1$s %2$s %3$s %4$s %5$s %6$s %7$s %8$s %9$s",
  "column" : [
    "host STRING",
    "identity STRING",
    "user STRING",
    "time STRING",
    "request STRING",
    "status STRING",
    "size STRING",
    "referer STRING",
    "agent STRING"
  ]
}
```

Sintassi

Campi opzionali	Description	Tipo di slot
column	Il nome della colonna con il tipo di dati specificato da ogni campo per i dati descritti da questo nodo di dati. Ex: hostname STRING Per più valori, utilizzare i nomi delle colonne e i tipi di dati separati da uno spazio.	Stringa
inputRegex	L'espressione regolare per analizzare un file di input S3. inputRegex fornisce un modo per recuperare colonne da dati relativamente non strutturati in un file.	Stringa

Campi opzionali	Description	Tipo di slot
outputFormat	I campi delle colonne recuperati da inputRegEx, ma a cui si fa riferimento come %1\$s %2\$s utilizzando la sintassi del formattatore Java.	Stringa
parent	Padre dell'oggetto corrente da cui saranno ereditati gli slot.	Oggetto di riferimento, ad esempio «parent»: {"ref":» Id "} myBaseObject

Campi Runtime	Description	Tipo di slot
@version	Versione della pipeline con cui l'oggetto è stato creato.	Stringa

Campi di sistema	Description	Tipo di slot
@error	Errore che descrive il formato oggetto errato	Stringa
@pipelineId	L'id della pipeline a cui appartiene questo oggetto	Stringa
@sphere	La sfera di un oggetto indica la propria posizione nel ciclo di vita: i Component Objects generano Instance Objects che eseguono Attempt Objects	Stringa

Formato dei dati TSV

Un formato di dati separati da virgola in cui il separatore di colonna è un carattere tab e il separatore di record è un carattere in una nuova riga.

Esempio

Di seguito è illustrato un esempio di questo tipo di oggetto.

```
{
  "id" : "MyOutputDataType",
  "type" : "TSV",
  "column" : [
    "Name STRING",
    "Score INT",
    "DateOfBirth TIMESTAMP"
  ]
}
```

Sintassi

Campi opzionali	Description	Tipo di slot
column	Il nome della colonna e il tipo di dati per i dati descritti da questo nodo di dati. Ad esempio "Name STRING" denota una colonna denominata Name con campi di tipo dati STRING. Separare più coppie di nomi di colonne e tipi di dati con virgole (come illustrato nell'esempio).	Stringa
columnSeparator	Carattere che separa i campi in una colonna dai campi nella colonna successiva. Impostazione predefinita su '\t'.	Stringa
escapeChar	Carattere, ad esempio "\", che indica al parser di ignorare il carattere successivo.	Stringa
parent	Padre dell'oggetto corrente da cui vengono ereditati gli slot.	Oggetto di riferimento, ad esempio «parent»: {"ref":» myBaseObject Id "}

Campi opzionali	Description	Tipo di slot
recordSeparator	Carattere che separa i record. Impostazione predefinita su '\n'.	Stringa

Campi Runtime	Description	Tipo di slot
@version	Versione della pipeline con cui l'oggetto è stato creato.	Stringa

Campi di sistema	Description	Tipo di slot
@error	Errore che descrive il formato oggetto errato.	Stringa
@pipelineId	L'ID della pipeline a cui appartiene questo oggetto.	Stringa
@sphere	La sfera di un oggetto indica la propria posizione nel ciclo di vita: i Component Objects generano Instance Objects che eseguono Attempt Objects.	Stringa

Azioni

Di seguito sono elencati gli oggetti attività AWS Data Pipeline:

Oggetti

- [SnsAlarm](#)
- [Interruzione](#)

SnsAlarm

Invia un messaggio di notifica Amazon SNS quando un'attività fallisce o termina con successo.

Esempio

Di seguito è illustrato un esempio di questo tipo di oggetto. I valori per `node.input` e `node.output` provengono dal nodo di dati o da un'attività che fa riferimento a questo oggetto nel relativo campo `onSuccess`.

```
{
  "id" : "SuccessNotify",
  "name" : "SuccessNotify",
  "type" : "SnsAlarm",
  "topicArn" : "arn:aws:sns:us-east-1:28619EXAMPLE:ExampleTopic",
  "subject" : "COPY SUCCESS: #{node.@scheduledStartTime}",
  "message" : "Files were copied from #{node.input} to #{node.output}."
}
```

Sintassi

Campi obbligatori	Description	Tipo di slot
message	Il testo del corpo della notifica Amazon SNS.	Stringa
role	Il ruolo IAM da utilizzare per creare l'allarme Amazon SNS.	Stringa
subject	L'oggetto del messaggio di notifica di Amazon SNS.	Stringa
topicArn	L'ARN dell'argomento Amazon SNS di destinazione per il messaggio.	Stringa

Campi opzionali	Description	Tipo di slot
parent	Padre dell'oggetto corrente da cui saranno ereditati gli slot.	Oggetto di riferimento, ad esempio «parent»: <code>{"ref":» myBaseObject Id ""}</code>

Campi Runtime	Description	Tipo di slot
nodo	Il nodo per il quale viene eseguita questa azione.	Oggetto di riferimento, ad esempio «node»: {"ref":» myRunnableObject Id "}
@version	Versione della pipeline con cui l'oggetto è stato creato.	Stringa

Campi di sistema	Description	Tipo di slot
@error	Errore che descrive il formato oggetto errato.	Stringa
@pipelineId	L'id della pipeline a cui appartiene questo oggetto.	Stringa
@sphere	La sfera di un oggetto indica la propria posizione nel ciclo di vita: i Component Objects generano Instance Objects che eseguono Attempt Objects.	Stringa

Interruzione

Azione per attivare l'annullamento di un'attività, una risorsa o un nodo di dati in sospeso o non terminati. AWS Data Pipeline tenta di inserire l'attività, la risorsa o il nodo dati nello stato CANCELLED se non si avvia per il valore `lateAfterTimeout`.

Non è possibile terminare azioni che includono risorse `onSuccess`, `OnFail` o `onLateAction`.

Esempio

Di seguito è illustrato un esempio di questo tipo di oggetto. In questo esempio, il campo `onLateAction` di `MyActivity` contiene un riferimento all'azione `DefaultAction1`. Quando si fornisce un'azione per `onLateAction`, è necessario fornire un valore `lateAfterTimeout` per

indicare il periodo di tempo dall'inizio programmato della pipeline dopo il quale l'attività è considerata in ritardo.

```
{
  "name" : "MyActivity",
  "id" : "DefaultActivity1",
  "schedule" : {
    "ref" : "MySchedule"
  },
  "runsOn" : {
    "ref" : "MyEmrCluster"
  },
  "lateAfterTimeout" : "1 Hours",
  "type" : "EmrActivity",
  "onLateAction" : {
    "ref" : "DefaultAction1"
  },
  "step" : [
    "s3://myBucket/myPath/myStep.jar,firstArg,secondArg",
    "s3://myBucket/myPath/myOtherStep.jar,anotherArg"
  ]
},
{
  "name" : "TerminateTasks",
  "id" : "DefaultAction1",
  "type" : "Terminate"
}
```

Sintassi

Campi opzionali	Description	Tipo di slot
parent	Padre dell'oggetto corrente da cui vengono ereditati gli slot.	Oggetto di riferimento, ad esempio «parent»: {"ref":» myBaseObject Id "}

Campi Runtime	Description	Tipo di slot
nodo	Il nodo per il quale viene eseguita questa azione.	Oggetto di riferimento, ad esempio «node»: {"ref":» myRunnableObject Id "}
@version	Versione della pipeline con cui l'oggetto è stato creato.	Stringa

Campi di sistema	Description	Tipo di slot
@error	Errore che descrive il formato oggetto errato.	Stringa
@pipelineId	L'ID della pipeline a cui appartiene questo oggetto.	Stringa
@sphere	La sfera di un oggetto indica la propria posizione nel ciclo di vita: i Component Objects generano Instance Objects che eseguono Attempt Objects.	Stringa

Pianificazione

Definisce la temporizzazione di un evento programmato, ad esempio quando si esegue un'attività.

Note

Quando il tempo di inizio di un programma è nel passato, AWS Data Pipeline esegue il backfill della pipeline e inizia la pianificazione delle esecuzioni subito all'inizio del tempo di avvio specificato. Per il test/lo sviluppo, utilizzare un intervallo relativamente breve. In caso contrario, AWS Data Pipeline tenta di mettere in coda e di pianificare tutte le esecuzioni della pipeline per tale intervallo. AWS Data Pipeline tenta di prevenire backfill accidentali se la componente della pipeline `scheduledStartTime` è precedente di 1 giorno bloccando l'attivazione della pipeline.

Esempi

Di seguito è illustrato un esempio di questo tipo di oggetto. Definisce una pianificazione per ogni ora, a partire da 00:00:00 il 2012-09-01 e finisce alle 00:00:00 il 2012-10-01. Il primo periodo termina alle 01:00:00 il 2012-09-01.

```
{
  "id" : "Hourly",
  "type" : "Schedule",
  "period" : "1 hours",
  "startDateTime" : "2012-09-01T00:00:00",
  "endDateTime" : "2012-10-01T00:00:00"
}
```

La pipeline seguente inizierà al `FIRST_ACTIVATION_DATE_TIME` e verrà eseguita ogni ora fino alle 22:00:00 il 2014-04-25.

```
{
  "id": "SchedulePeriod",
  "name": "SchedulePeriod",
  "startAt": "FIRST_ACTIVATION_DATE_TIME",
  "period": "1 hours",
  "type": "Schedule",
  "endDateTime": "2014-04-25T22:00:00"
}
```

La pipeline seguente inizierà al `FIRST_ACTIVATION_DATE_TIME` e verrà eseguita ogni ora e completata dopo tre occorrenze.

```
{
  "id": "SchedulePeriod",
  "name": "SchedulePeriod",
  "startAt": "FIRST_ACTIVATION_DATE_TIME",
  "period": "1 hours",
  "type": "Schedule",
  "occurrences": "3"
}
```

La seguente pipeline inizierà alle 22:00:00 il 2014-04-25, verrà eseguita ogni ora e terminerà dopo tre occorrenze.

```
{
  "id": "SchedulePeriod",
  "name": "SchedulePeriod",
  "startDateTime": "2014-04-25T22:00:00",
  "period": "1 hours",
  "type": "Schedule",
  "occurrences": "3"
}
```

On-demand utilizzando l'oggetto predefinito

```
{
  "name": "Default",
  "resourceRole": "DataPipelineDefaultResourceRole",
  "role": "DataPipelineDefaultRole",
  "scheduleType": "ondemand"
}
```

On-demand con un oggetto di pianificazione esplicita

```
{
  "name": "Default",
  "resourceRole": "DataPipelineDefaultResourceRole",
  "role": "DataPipelineDefaultRole",
  "scheduleType": "ondemand"
},
{
  "name": "DefaultSchedule",
  "type": "Schedule",
  "id": "DefaultSchedule",
  "period": "ONDEMAND_PERIOD",
  "startAt": "ONDEMAND_ACTIVATION_TIME"
},
```

I seguenti esempi illustrano come una pianificazione può essere ereditata dall'oggetto predefinito, essere esplicitamente impostata per l'oggetto o derivare da un riferimento padre:

Pianificazione ereditata da un oggetto predefinito

```
{
  "objects": [
```

```

{
  "id": "Default",
  "failureAndRerunMode": "cascade",
  "resourceRole": "DataPipelineDefaultResourceRole",
  "role": "DataPipelineDefaultRole",
  "pipelineLogUri": "s3://myLogsbucket",
  "scheduleType": "cron",
  "schedule": {
    "ref": "DefaultSchedule"
  }
},
{
  "type": "Schedule",
  "id": "DefaultSchedule",
  "occurrences": "1",
  "period": "1 Day",
  "startAt": "FIRST_ACTIVATION_DATE_TIME"
},
{
  "id": "A_Fresh_NewEC2Instance",
  "type": "Ec2Resource",
  "terminateAfter": "1 Hour"
},
{
  "id": "ShellCommandActivity_HelloWorld",
  "runsOn": {
    "ref": "A_Fresh_NewEC2Instance"
  },
  "type": "ShellCommandActivity",
  "command": "echo 'Hello World!'"
}
]
}

```

Pianificazione esplicita sull'oggetto

```

{
  "objects": [
    {
      "id": "Default",
      "failureAndRerunMode": "cascade",
      "resourceRole": "DataPipelineDefaultResourceRole",
      "role": "DataPipelineDefaultRole",

```

```

    "pipelineLogUri": "s3://myLogsbucket",
    "scheduleType": "cron"
  },
  {
    "type": "Schedule",
    "id": "DefaultSchedule",
    "occurrences": "1",
    "period": "1 Day",
    "startAt": "FIRST_ACTIVATION_DATE_TIME"
  },
  {
    "id": "A_Fresh_NewEC2Instance",
    "type": "Ec2Resource",
    "terminateAfter": "1 Hour"
  },
  {
    "id": "ShellCommandActivity_HelloWorld",
    "runsOn": {
      "ref": "A_Fresh_NewEC2Instance"
    },
    "schedule": {
      "ref": "DefaultSchedule"
    },
    "type": "ShellCommandActivity",
    "command": "echo 'Hello World!'"
  }
]
}

```

Pianificazione da un riferimento padre

```

{
  "objects": [
    {
      "id": "Default",
      "failureAndRerunMode": "cascade",
      "resourceRole": "DataPipelineDefaultResourceRole",
      "role": "DataPipelineDefaultRole",
      "pipelineLogUri": "s3://myLogsbucket",
      "scheduleType": "cron"
    },
  ],
}

```

```

{
  "id": "parent1",
  "schedule": {
    "ref": "DefaultSchedule"
  }
},
{
  "type": "Schedule",
  "id": "DefaultSchedule",
  "occurrences": "1",
  "period": "1 Day",
  "startAt": "FIRST_ACTIVATION_DATE_TIME"
},
{
  "id": "A_Fresh_NewEC2Instance",
  "type": "Ec2Resource",
  "terminateAfter": "1 Hour"
},
{
  "id": "ShellCommandActivity_HelloWorld",
  "runsOn": {
    "ref": "A_Fresh_NewEC2Instance"
  },
  "parent": {
    "ref": "parent1"
  },
  "type": "ShellCommandActivity",
  "command": "echo 'Hello World!'"
}
]
}

```

Sintassi

Campi obbligatori	Description	Tipo di slot
punto	Frequenza di esecuzione della pipeline. Il formato è "N [minuti ore giorni settimane mesi]", dove N è un numero seguito da uno degli specificatori di tempo. Ad esempio, "15 minuti" esegue la pipeline ogni 15 minuti. La durata	Periodo

Campi obbligatori	Description	Tipo di slot
	minima è pari a 15 minuti, mentre la durata massima è di 3 anni.	
Gruppo richiesto (uno dei seguenti è obbligatorio)	Description	Tipo di slot
startAt	La data e l'ora in cui avviare l'esecuzione della pipeline programmata. Il valore valido è FIRST_ACTIVATION_DATE_TIME, che viene sostituito dalla creazione di una pipeline on demand.	Enumerazione
startDateTime	La data e l'ora di avvio delle esecuzioni programmate. È necessario utilizzare uno startDateTime o startAt ma non entrambi.	DateTime
Campi opzionali	Description	Tipo di slot
endDateTime	La data e l'ora in cui terminare le esecuzioni i programmate. Deve essere una data e un'ora successive al valore di startDateTime o startAt. Il comportamento predefinito è quello di pianificare l'esecuzione finché la pipeline non viene terminata.	DateTime
occorrenze	Il numero di volte in cui eseguire la pipeline quando viene attivata. Non è possibile utilizzare le occorrenze con. endDateTime	Numero intero
parent	Padre dell'oggetto corrente da cui saranno ereditati gli slot.	Oggetto di riferimento, ad esempio «parent»:

Campi opzionali	Description	Tipo di slot
		{"ref»:» myBaseObject Id "}
Campi Runtime	Description	Tipo di slot
@version	Versione della pipeline con cui l'oggetto è stato creato.	Stringa
Campi di sistema	Description	Tipo di slot
@error	Errore che descrive il formato oggetto errato	Stringa
@firstActivationTime	L'ora della creazione di oggetti.	DateTime
@pipelineId	L'id della pipeline a cui appartiene questo oggetto	Stringa
@sphere	La sfera di un oggetto indica la propria posizione nel ciclo di vita: i Component Objects generano Instance Objects che eseguono Attempt Objects	Stringa

Utilità

I seguenti oggetti della utility configurano altri oggetti della pipeline:

Argomenti

- [ShellScriptConfig](#)
- [EmrConfiguration](#)
- [Proprietà](#)

ShellScriptConfig

Da utilizzare con un'attività per eseguire uno script di shell per preActivityTask Config e postActivityTask Config. Questo oggetto è disponibile per [HadoopActivity](#), [HiveActivityHiveCopyActivity](#), e [PigActivity](#). Specificare un URI S3 e un elenco di argomenti per lo script.

Esempio

A ShellScriptConfig con argomenti:

```
{
  "id" : "ShellScriptConfig_1",
  "name" : "prescript",
  "type" : "ShellScriptConfig",
  "scriptUri": "s3://my-bucket/shell-cleanup.sh",
  "scriptArgument" : ["arg1","arg2"]
}
```

Sintassi

Questo oggetto include i campi seguenti.

Campi opzionali	Description	Tipo di slot
parent	Padre dell'oggetto corrente da cui vengono ereditati gli slot.	Oggetto di riferimento, ad esempio «parent»: {"ref»:» myBaseObject Id "}
scriptArgument	Un elenco di argomenti da utilizzare con lo script della shell.	Stringa
scriptUri	L'URI dello script in Amazon S3 che deve essere scaricato ed eseguito.	Stringa

Campi Runtime	Description	Tipo di slot
@version	Versione della pipeline con cui l'oggetto è stato creato.	Stringa
Campi di sistema	Description	Tipo di slot
@error	Errore che descrive il formato oggetto errato.	Stringa
@pipelineId	L'ID della pipeline a cui appartiene questo oggetto.	Stringa
@sphere	La sfera di un oggetto indica la propria posizione nel ciclo di vita: i Component Objects generano Instance Objects che eseguono Attempt Objects.	Stringa

EmrConfiguration

L'EmrConfiguration oggetto è la configurazione utilizzata per i cluster EMR con versioni 4.0.0 o successive. Le configurazioni (sotto forma di elenco) sono un parametro per la chiamata API. RunJobFlow L'API di configurazione per Amazon EMR richiede una classificazione e proprietà. AWS Data Pipeline utilizza EmrConfiguration con gli oggetti Property corrispondenti per configurare un'[EmrCluster](#) applicazione come Hadoop, Hive, Spark o Pig su cluster EMR avviati in un'esecuzione di pipeline. Poiché la configurazione può essere modificata solo per i nuovi cluster, non è possibile fornire un oggetto per le risorse esistenti. EmrConfiguration Per ulteriori informazioni, consulta <https://docs.aws.amazon.com/ElasticMapReduce/latest/ReleaseGuide/>.

Esempio

Il seguente oggetto di configurazione imposta le proprietà `io.file.buffer.size` e `fs.s3.block.size` in `core-site.xml`:

```
[
  {
    "classification": "core-site",
```

```
    "properties":
    {
      "io.file.buffer.size": "4096",
      "fs.s3.block.size": "67108864"
    }
  }
]
```

La definizione dell'oggetto della pipeline corrispondente utilizza un `EmrConfiguration` oggetto e un elenco di oggetti `Property` nel `property` campo:

```
{
  "objects": [
    {
      "name": "ReleaseLabelCluster",
      "releaseLabel": "emr-4.1.0",
      "applications": ["spark", "hive", "pig"],
      "id": "ResourceId_I1mCc",
      "type": "EmrCluster",
      "configuration": {
        "ref": "coresite"
      }
    },
    {
      "name": "coresite",
      "id": "coresite",
      "type": "EmrConfiguration",
      "classification": "core-site",
      "property": [{
        "ref": "io-file-buffer-size"
      },
      {
        "ref": "fs-s3-block-size"
      }
    ],
    {
      "name": "io-file-buffer-size",
      "id": "io-file-buffer-size",
      "type": "Property",
      "key": "io.file.buffer.size",
      "value": "4096"
    },
  ],
}
```

```

    {
      "name": "fs-s3-block-size",
      "id": "fs-s3-block-size",
      "type": "Property",
      "key": "fs.s3.block.size",
      "value": "67108864"
    }
  ]
}

```

L'esempio seguente è una configurazione nidificata utilizzata per impostare l'ambiente Hadoop con la classificazione `hadoop-env`:

```

[
  {
    "classification": "hadoop-env",
    "properties": {},
    "configurations": [
      {
        "classification": "export",
        "properties": {
          "YARN_PROXYSERVER_HEAPSIZE": "2396"
        }
      }
    ]
  }
]

```

L'oggetto di definizione corrispondente della pipeline che utilizza questa configurazione è il seguente:

```

{
  "objects": [
    {
      "name": "ReleaseLabelCluster",
      "releaseLabel": "emr-4.0.0",
      "applications": ["spark", "hive", "pig"],
      "id": "ResourceId_I1mCc",
      "type": "EmrCluster",
      "configuration": {
        "ref": "hadoop-env"
      }
    },
    {

```

```

    "name": "hadoop-env",
    "id": "hadoop-env",
    "type": "EmrConfiguration",
    "classification": "hadoop-env",
    "configuration": {
      "ref": "export"
    }
  },
  {
    "name": "export",
    "id": "export",
    "type": "EmrConfiguration",
    "classification": "export",
    "property": {
      "ref": "yarn-proxyserver-heapsize"
    }
  },
  {
    "name": "yarn-proxyserver-heapsize",
    "id": "yarn-proxyserver-heapsize",
    "type": "Property",
    "key": "YARN_PROXYSERVER_HEAPSIZE",
    "value": "2396"
  },
]
}

```

L'esempio seguente modifica una proprietà specifica di Hive per un cluster EMR:

```

{
  "objects": [
    {
      "name": "hivesite",
      "id": "hivesite",
      "type": "EmrConfiguration",
      "classification": "hive-site",
      "property": [
        {
          "ref": "hive-client-timeout"
        }
      ]
    },
    {

```

```

    "name": "hive-client-timeout",
    "id": "hive-client-timeout",
    "type": "Property",
    "key": "hive.metastore.client.socket.timeout",
    "value": "2400s"
  }
]
}

```

Sintassi

Questo oggetto include i campi seguenti.

Campi obbligatori	Description	Tipo di slot
classificazione	Classificazione della configurazione.	Stringa

Campi opzionali	Description	Tipo di slot
configurazione	Sottoconfigurazione per questa configurazione.	Oggetto di riferimento, ad esempio «configuration»: {"ref":» Id "} myEmrConfiguration
parent	Padre dell'oggetto corrente da cui saranno ereditati gli slot.	Oggetto di riferimento, ad esempio «parent»: {"ref":» myBaseObject Id "}
property	Proprietà di configurazione.	Oggetto di riferimento, ad esempio «property»: {"ref":» myPropertyId «}

Campi Runtime	Description	Tipo di slot
@version	Versione della pipeline con cui l'oggetto è stato creato.	Stringa

Campi di sistema	Description	Tipo di slot
@error	Errore che descrive il formato oggetto errato	Stringa
@pipelineId	L'id della pipeline a cui appartiene questo oggetto	Stringa
@sphere	La sfera di un oggetto indica la propria posizione nel ciclo di vita: i Component Objects generano Instance Objects che eseguono Attempt Objects	Stringa

Vedi anche

- [EmrCluster](#)
- [Proprietà](#)
- [Guida al rilascio di Amazon EMR](#)

Proprietà

Una singola proprietà chiave-valore da utilizzare con un EmrConfiguration oggetto.

Esempio

La seguente definizione della pipeline mostra un EmrConfiguration oggetto e gli oggetti Property corrispondenti per lanciare un: EmrCluster

```
{  
  "objects": [  

```

```
{
  "name": "ReleaseLabelCluster",
  "releaseLabel": "emr-4.1.0",
  "applications": ["spark", "hive", "pig"],
  "id": "ResourceId_I1mCc",
  "type": "EmrCluster",
  "configuration": {
    "ref": "coresite"
  }
},
{
  "name": "coresite",
  "id": "coresite",
  "type": "EmrConfiguration",
  "classification": "core-site",
  "property": [{
    "ref": "io-file-buffer-size"
  },
  {
    "ref": "fs-s3-block-size"
  }
],
{
  "name": "io-file-buffer-size",
  "id": "io-file-buffer-size",
  "type": "Property",
  "key": "io.file.buffer.size",
  "value": "4096"
},
{
  "name": "fs-s3-block-size",
  "id": "fs-s3-block-size",
  "type": "Property",
  "key": "fs.s3.block.size",
  "value": "67108864"
}
]
```

Sintassi

Questo oggetto include i campi seguenti.

Campi obbligatori	Description	Tipo di slot
Chiave	key	Stringa
value	value	Stringa

Campi opzionali	Description	Tipo di slot
parent	Padre dell'oggetto corrente da cui vengono ereditati gli slot.	Oggetto di riferimento, ad esempio «parent»: {"ref":» myBaseObject Id "}

Campi Runtime	Description	Tipo di slot
@version	Versione della pipeline con cui l'oggetto è stato creato.	Stringa

Campi di sistema	Description	Tipo di slot
@error	Errore che descrive il formato oggetto errato.	Stringa
@pipelineId	L'ID della pipeline a cui appartiene questo oggetto.	Stringa
@sphere	La sfera di un oggetto indica la propria posizione nel ciclo di vita: i Component Objects generano Instance Objects che eseguono Attempt Objects.	Stringa

Vedi anche

- [EmrCluster](#)
- [EmrConfiguration](#)
- [Guida al rilascio di Amazon EMR](#)

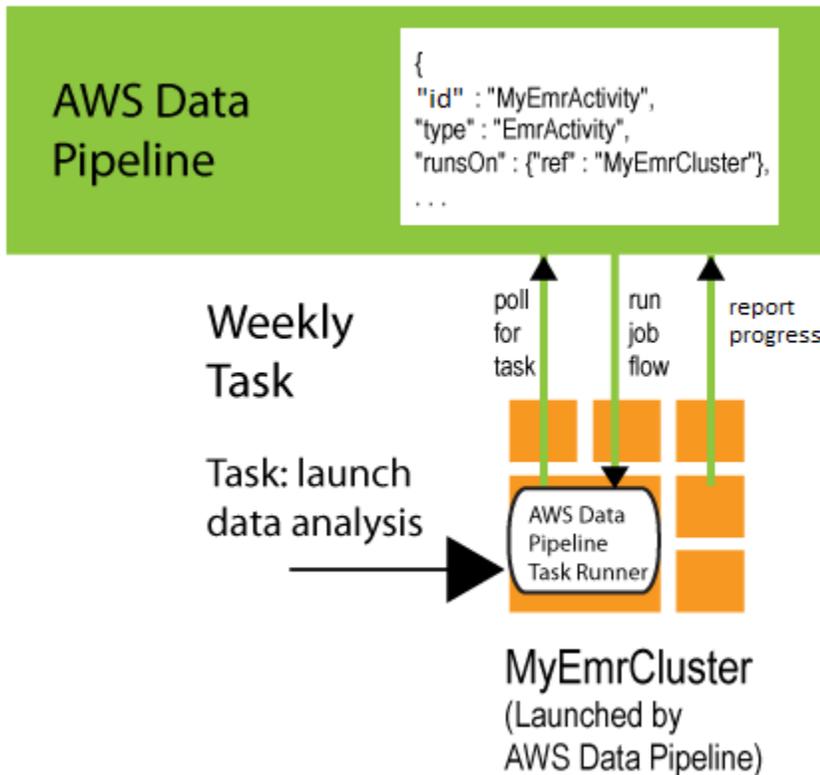
Utilizzo di Task Runner

Task Runner è un'applicazione task agent che effettua sondaggi AWS Data Pipeline per individuare le attività pianificate e le esegue su istanze Amazon EC2, cluster Amazon EMR o altre risorse computazionali, segnalandone lo stato man mano che lo fa. A seconda dell'applicazione, è possibile scegliere di:

- Consentire a AWS Data Pipeline di installare e gestire una o più applicazioni Task Runner per te. Quando viene attivata una pipeline, viene creato automaticamente il valore predefinito `Ec2Instance` o `EmrCluster` oggetto a cui fa riferimento un campo `runsOn` dell'attività. AWS Data Pipeline si occupa dell'installazione di Task Runner su un'istanza EC2 o sul nodo master di un cluster EMR. In questo modello, AWS Data Pipeline è in grado di eseguire la maggior parte delle operazioni di gestione dei cluster o delle istanze per l'utente.
- Eseguire tutta o parte di una pipeline su risorse gestite dall'utente. Le risorse potenziali includono un'istanza Amazon EC2 di lunga durata, un cluster Amazon EMR o un server fisico. È possibile installare un task runner (che può essere Task Runner o un task agent personalizzato del proprio dispositivo) quasi ovunque, a condizione che possa comunicare con il servizio AWS Data Pipeline Web. In questo schema, si assume il controllo quasi completo su quali risorse vengono utilizzate e su come vengono gestite e è necessario installare e configurare manualmente Task Runner. Per eseguire questa operazione, utilizzare le procedure di questa sezione, come descritto in [Esecuzione di lavori su risorse esistenti utilizzando Task Runner](#).

Task Runner AWS Data Pipeline su risorse gestite

Quando una risorsa viene avviata e gestita da AWS Data Pipeline, il servizio Web installa automaticamente Task Runner su quella risorsa per elaborare le attività nella pipeline. Si specifica una risorsa computazionale (un'istanza Amazon EC2 o un cluster Amazon EMR) per il `runsOn` campo di un oggetto di attività. Quando AWS Data Pipeline avvia questa risorsa, installa Task Runner su quella risorsa e la configura per elaborare tutti gli oggetti di attività il cui `runsOn` campo è impostato su quella risorsa. Quando AWS Data Pipeline termina la risorsa, i log di Task Runner vengono pubblicati in una posizione Amazon S3 prima che si spenga.



Ad esempio, se si utilizza la `EmrActivity` in una pipeline e si specifica una risorsa `EmrCluster` nel campo `runsOn`. Quando AWS Data Pipeline elabora tale attività, avvia un cluster Amazon EMR e installa Task Runner sul nodo master. Questo Task Runner elabora quindi le attività per le attività il cui `runsOn` campo è impostato su quell'`EmrCluster` oggetto. Il seguente estratto da una definizione di pipeline mostra questa relazione tra due oggetti.

```
{
  "id" : "MyEmrActivity",
  "name" : "Work to perform on my data",
  "type" : "EmrActivity",
  "runsOn" : {"ref" : "MyEmrCluster"},
  "preStepCommand" : "scp remoteFiles localFiles",
  "step" : "s3://myBucket/myPath/myStep.jar,firstArg,secondArg",
  "step" : "s3://myBucket/myPath/myOtherStep.jar,anotherArg",
  "postStepCommand" : "scp localFiles remoteFiles",
  "input" : {"ref" : "MyS3Input"},
  "output" : {"ref" : "MyS3Output"}
},
{
  "id" : "MyEmrCluster",
  "name" : "EMR cluster to perform the work",
```

```
"type" : "EmrCluster",
"hadoopVersion" : "0.20",
"keypair" : "myKeyPair",
"masterInstanceType" : "m1.xlarge",
"coreInstanceType" : "m1.small",
"coreInstanceCount" : "10",
"taskInstanceType" : "m1.small",
"taskInstanceCount": "10",
"bootstrapAction" : "s3://elasticmapreduce/libs/ba/configure-hadoop, arg1, arg2, arg3",
"bootstrapAction" : "s3://elasticmapreduce/libs/ba/configure-other-stuff, arg1, arg2"
}
```

Per informazioni ed esempi di esecuzione di questa attività, vedi [EmrActivity](#).

Se in una pipeline sono presenti più risorse AWS Data Pipeline gestite, Task Runner viene installato su ciascuna di esse e tutte analizzano AWS Data Pipeline le attività da elaborare.

Esecuzione di lavori su risorse esistenti utilizzando Task Runner

Puoi installare Task Runner su risorse computazionali che gestisci, come un'istanza Amazon EC2 o un server o una workstation fisici. Task Runner può essere installato ovunque, su qualsiasi hardware o sistema operativo compatibile, a condizione che possa comunicare con il servizio AWS Data Pipeline web.

Questo approccio può essere utile se, ad esempio, si desidera utilizzare AWS Data Pipeline per elaborare i dati memorizzati all'interno del firewall aziendale. Installando Task Runner su un server della rete locale, è possibile accedere al database locale in modo sicuro e quindi eseguire un sondaggio AWS Data Pipeline per l'attività successiva da eseguire. Quando AWS Data Pipeline termina l'elaborazione o elimina la pipeline, l'istanza Task Runner rimane in esecuzione sulla risorsa di calcolo fino a quando non la chiudi manualmente. I log di Task Runner persistono al termine dell'esecuzione della pipeline.

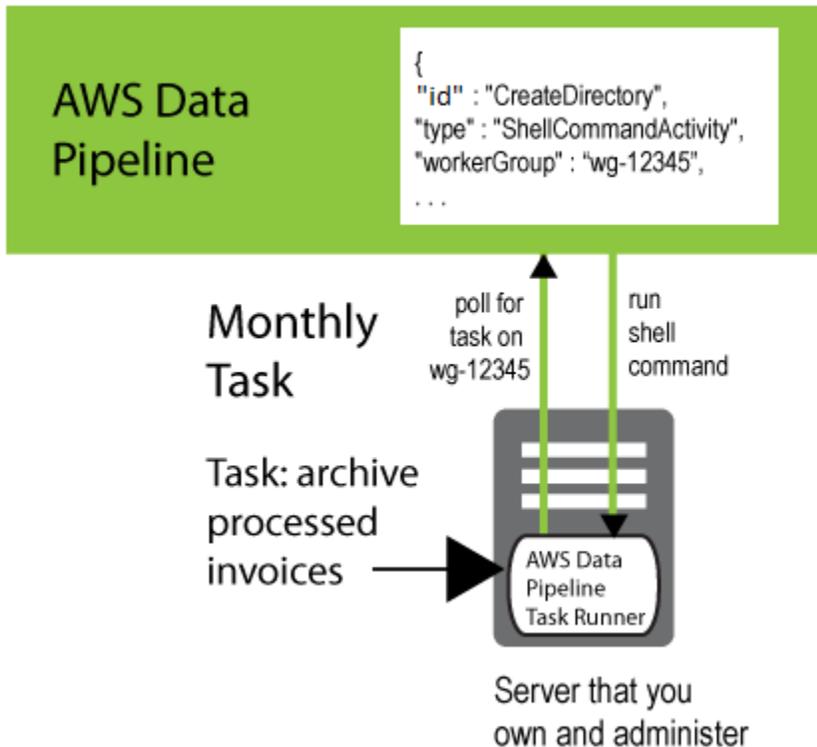
Per utilizzare Task Runner su una risorsa che gestisci, devi prima scaricare Task Runner e quindi installarlo sulla tua risorsa di calcolo, utilizzando le procedure in questa sezione.

Note

Puoi installare Task Runner solo su Linux, UNIX o macOS. Task Runner non è supportato nel sistema operativo Windows.

Per utilizzare Task Runner 2.0, la versione minima di Java necessaria è 1.7.

Per connettere un Task Runner che hai installato alle attività della pipeline che deve elaborare, aggiungi un `workerGroup` campo all'oggetto e configura Task Runner in modo che esegua il sondaggio per quel valore del gruppo di lavoro. A tale scopo, si passa la stringa del gruppo di lavoro come parametro (ad esempio `--workerGroup=wg-12345`) quando si esegue il file JAR Task Runner.



```
{
  "id" : "CreateDirectory",
  "type" : "ShellCommandActivity",
  "workerGroup" : "wg-12345",
  "command" : "mkdir new-directory"
}
```

Installazione di Task Runner

Questa sezione spiega come installare e configurare Task Runner e i relativi prerequisiti. L'installazione è un semplice processo manuale.

Per installare Task Runner

1. Task Runner richiede le versioni Java 1.6 o 1.8. Per determinare se Java è installato e la versione in esecuzione, utilizzare il comando seguente:

```
java -version
```

Se sul computer non è installato Java 1.6 o 1.8, scarica una di queste versioni da <http://www.oracle.com/technetwork/java/index.html>. Scaricare e installare Java, quindi procedere con il passaggio successivo.

2. Scaricare `TaskRunner-1.0.jar` da <https://s3.amazonaws.com/datapipeline-us-east-1/us-east-1/software/latest/TaskRunner-1.0.jar> e copiarlo in una cartella sulla risorsa informatica di destinazione. Per i cluster Amazon EMR che eseguono `EmrActivity` attività, installa Task Runner sul nodo master del cluster.
3. Quando si utilizza Task Runner per connettersi al servizio AWS Data Pipeline Web per elaborare i comandi, gli utenti devono accedere programmaticamente a un ruolo che dispone delle autorizzazioni per creare o gestire pipeline di dati. Per ulteriori informazioni, consulta [Concessione dell'accesso programmatico](#).
4. Task Runner si connette al servizio AWS Data Pipeline Web tramite HTTPS. Se si sta usando una risorsa AWS, assicurarsi che HTTPS sia abilitato nella tabella di routing e nella sottorete ACL. Se si sta usando un firewall o un proxy, assicurarsi che la porta 443 sia aperta.

(Facoltativo) Concessione dell'accesso a Task Runner ad Amazon RDS

Amazon RDS ti permette di controllare l'accesso alle istanze DB utilizzando i gruppi di sicurezza del database (i gruppi di sicurezza del database). Un gruppo di sicurezza DB si comporta come un firewall, controllando l'accesso di rete all'istanza database. Per impostazione predefinita, l'accesso alla rete è disattivato per le istanze database. È necessario modificare i gruppi di sicurezza del database per consentire a Task Runner di accedere alle istanze Amazon RDS. Task Runner ottiene l'accesso ad Amazon RDS dall'istanza su cui viene eseguito, quindi gli account e i gruppi di sicurezza che aggiungi alla tua istanza Amazon RDS dipendono da dove installi Task Runner.

Per concedere l'accesso a Task Runner in EC2-Classik

1. Apri la console Amazon RDS.
2. Nel riquadro di navigazione scegliere Instances (Istanze) e quindi selezionare l'istanza database.

3. In Security and Network (Sicurezza e Network), selezionare il gruppo di sicurezza che apre la pagina relativa ai Security Groups (Gruppi di sicurezza) con questo gruppo di sicurezza di database selezionato. Selezionare l'icona dei dettagli per il gruppo di sicurezza DB.
4. In Security Group Details (Dettagli gruppo di sicurezza), creare una regola con il Connection Type (Tipo di connessione) e i Details (Dettagli) appropriati. Questi campi dipendono dalla posizione in cui Task Runner è in esecuzione, come descritto qui:
 - Ec2Resource
 - Connection Type (Tipo di connessione): EC2 Security Group
 - Dettagli: *my-security-group-name* (il nome del gruppo di sicurezza creato per l'istanza EC2)
 - EmrResource
 - Connection Type (Tipo di connessione): EC2 Security Group
 - Dettagli (Dettagli): ElasticMapReduce-master
 - Connection Type (Tipo di connessione): EC2 Security Group
 - Dettagli (Dettagli): ElasticMapReduce-slave
 - Ambiente locale (in locale)
 - Connection Type (Tipo di connessione): CIDR/IP:
 - Dettagli: *my-ip-address* (l'indirizzo IP del computer o l'intervallo di indirizzi IP della rete, se il computer è protetto da un firewall)
5. Fare clic su Add (Aggiungi).

Per concedere l'accesso a Task Runner in EC2-VPC

1. Apri la console Amazon RDS.
2. Nel pannello di navigazione, seleziona Instances (Istanze).
3. Selezionare l'icona dei dettagli per l'istanza database. In Sicurezza e rete, apri il link al gruppo di sicurezza, che ti porta alla console Amazon EC2. Se si utilizza il vecchio progetto della console per i gruppi di sicurezza, passare al nuovo progetto della console selezionando l'icona visualizzata nella parte superiore della pagina della console.

4. Nella scheda Inbound (In entrata), scegli Edit (Modifica), Add Rule (Aggiungi regola). Specificare la porta del database utilizzata quando è stata avviata l'istanza database. La fonte dipende da dove è in esecuzione Task Runner, come descritto qui:
 - `Ec2Resource`
 - `my-security-group-id` (l'ID del gruppo di sicurezza creato per l'istanza EC2)
 - `EmrResource`
 - `master-security-group-id` (l'ID del gruppoElasticMapReduce-master di sicurezza)
 - `slave-security-group-id` (l'ID del gruppoElasticMapReduce-slave di sicurezza)
 - Ambiente locale (in locale)
 - `ip-address` (l'indirizzo IP del computer o l'intervallo di indirizzi IP della rete, se il computer è protetto da un firewall)
5. Fare clic su Save (Salva).

Avvio di Task Runner

In una nuova finestra del prompt dei comandi impostata sulla directory in cui hai installato Task Runner, avvia Task Runner con il seguente comando.

```
java -jar TaskRunner-1.0.jar --config ~/credentials.json --workerGroup=myWorkerGroup --region=MyRegion --logUri=s3://mybucket/foldername
```

L'opzione `--config` punta al file delle credenziali.

L'opzione `--workerGroup` specifica il nome del gruppo di lavoratori, che deve essere lo stesso valore specificato nella pipeline per le attività da elaborare.

L'opzione `--region` specifica la regione del servizio da cui prendere le operazioni da eseguire.

L'opzione `--logUri` viene utilizzata per trasferire i log compressi in una posizione in Amazon S3.

Quando Task Runner è attivo, stampa il percorso in cui vengono scritti i file di registro nella finestra del terminale. Di seguito è riportato un esempio.

```
Logging to /Computer_Name/.../output/logs
```

Task Runner deve essere eseguito non collegato alla shell di login. Se si sta usando un'applicazione terminale per connettersi al computer, potrebbe essere necessario utilizzare una utility come `nohup`

o schermo per evitare di uscire dall'applicazione Task Runner al momento della disconnessione. Per ulteriori informazioni sulle opzioni delle righe di comando, consulta [Opzioni di configurazione di Task Runner](#).

Verifica della registrazione di Task Runner

Il modo più semplice per verificare che Task Runner funzioni è verificare se sta scrivendo file di registro. Task Runner scrive i file di registro ogni ora nella `directoryoutput/logs`, sotto la directory in cui è installato Task Runner. Il nome del file è `Task Runner.log.YYYY-MM-DD-HH`, dove HH viene eseguito da mezzanotte alle 23:00, in UDT. Per risparmiare spazio di archiviazione, i file di log con più di otto ore sono compressi con GZip.

Thread e prerequisiti di Task Runner

Task Runner utilizza un pool di thread per ciascuna delle attività, delle attività e delle precondizioni. L'impostazione predefinita per `--tasks` è 2, il che significa che non ci sono due thread allocati dal pool di attività e ogni thread esegue il polling del servizio AWS Data Pipeline per le nuove attività. Pertanto, `--tasks` è un attributo di ottimizzazione delle prestazioni che può essere utilizzato per ottimizzare il throughput della pipeline.

La logica di ripetizione della pipeline per le precondizioni si verifica in Task Runner. Vengono assegnati due thread di precondizione AWS Data Pipeline per il sondaggio degli oggetti precondizionati. Task Runner rispetta i campi `retryDelay` e `preconditionTimeout` dell'oggetto precondizione definiti in base alle precondizioni.

In molti casi, diminuendo il timeout del polling della precondizione e il numero di nuovi tentativi è possibile migliorare le prestazioni dell'applicazione. Analogamente, le applicazioni con precondizioni di lunga durata potrebbero avere bisogno di un aumento dei valori relativi a timeout e nuovi tentativi. Per ulteriori informazioni sugli oggetti delle precondizioni, consulta [Precondizioni](#).

Opzioni di configurazione di Task Runner

Queste sono le opzioni di configurazione disponibili dalla riga di comando all'avvio di Task Runner.

Parametri della riga di comando	Descrizione
<code>--help</code>	Informazioni di aiuto della riga di comando. Esempio: <code>Java -jar TaskRunner-1.0.jar --help</code>
<code>--config</code>	Percorso e nome del file <code>credentials.json</code> .
<code>--accessId</code>	L'ID della chiave di AWS accesso che Task Runner può utilizzare per effettuare richieste. Le opzioni <code>--secretKey</code> e <code>--accessID</code> forniscono un'alternativa all'utilizzo di un file <code>credentials.json</code> . Se viene fornito anche un file <code>credentials.json</code> , le opzioni <code>--accessID</code> e <code>--secretKey</code> hanno la precedenza.
<code>--secretKey</code>	La tua chiave AWS segreta che Task Runner può utilizzare per effettuare richieste. Per ulteriori informazioni, consulta <code>--accessID</code> .
<code>--endpoint</code>	Un endpoint è un URL che rappresenta il punto di partenza per un servizio Web. L'endpoint del servizio AWS Data Pipeline nella regione in cui si effettuano le richieste. Facoltativo. In generale, è sufficiente specificare una regione e non è necessario impostare l'endpoint. Per un elenco degli endpoint e delle regioni AWS Data Pipeline disponibili, consulta Regioni ed endpoint di AWS Data Pipeline in Riferimenti generali di AWS.
<code>--workerGroup</code>	Il nome del gruppo di lavoratori per cui Task Runner recupera il lavoro. Campo obbligatorio. Quando Task Runner esegue un sondaggio sul servizio Web, utilizza le credenziali fornite

Parametri della riga di comando	Descrizione
	<p>e il valore di <code>workerGroup</code> per selezionare quali (se presenti) attività recuperare. Puoi usare qualsiasi nome che sia significativo per te; l'unico requisito è che la stringa corrisponda tra Task Runner e le attività della pipeline corrispondenti. Il nome del gruppo di lavoratori è associato a una regione. Anche se esistono nomi di gruppi di lavoro identici in altre regioni, Task Runner ottiene sempre le attività dalla regione specificata in <code>--region</code>.</p>
<code>--taskrunnerId</code>	<p>L'ID del runner dell'attività da utilizzare per segnalare l'avanzamento. Facoltativo.</p>
<code>--output</code>	<p>La directory Task Runner per i file di output dei log. Facoltativo. I file di registro vengono archiviati in una directory locale fino a quando non vengono inviati ad Amazon S3. Questa opzione sostituisce la directory di default.</p>
<code>--region</code>	<p>La regione da utilizzare. L'impostazione della regione è facoltativa, ma è sempre consigliata. Se non si specifica la regione, Task Runner recupera le attività dall'area di servizio predefinita, <code>us-east-1</code>.</p> <p>Altre regioni supportate sono: <code>eu-west-1</code>, <code>ap-northeast-1</code>, <code>ap-southeast-2</code>, <code>us-west-2</code>.</p>
<code>--logUri</code>	<p>Il percorso di destinazione di Amazon S3 in cui Task Runner esegue il backup dei file di registro ogni ora. Quando Task Runner termina, i log attivi nella directory locale vengono trasferiti nella cartella di destinazione di Amazon S3.</p>

Parametri della riga di comando	Descrizione
<code>--proxyHost</code>	L'host del proxy utilizzato dai client Task Runner per connettersi ai servizi AWS.
<code>--proxyPort</code>	Porta dell'host proxy utilizzata dai client Task Runner per connettersi ai servizi AWS.
<code>--proxyUsername</code>	Il nome utente per il proxy.
<code>--proxyPassword</code>	La password per il proxy.
<code>--proxyDomain</code>	Il nome del dominio di Windows per il proxy NTLM.
<code>--proxyWorkstation</code>	Il nome della workstation di Windows per il proxy NTLM.

Utilizzo di Task Runner con un proxy

Se si sta usando un host proxy, è possibile specificarne la [configurazione](#) quando si richiama Task Runner o si imposta la variabile di ambiente, `HTTPS_PROXY`. La variabile di ambiente usata con Task Runner accetta la stessa configurazione utilizzata per l'[interfaccia a riga di comando di AWS](#).

Task Runner e AMI personalizzate

Quando specifichi un `Ec2Resource` oggetto per la tua pipeline, AWS Data Pipeline crea un'istanza EC2 per te, utilizzando un'AMI che installa e configura Task Runner per te. Un tipo di istanza compatibile con PV è obbligatorio in questo caso. In alternativa, puoi creare un'AMI personalizzata con Task Runner e quindi specificare l'ID di questa AMI utilizzando il `imageId` campo dell'`Ec2Resource` oggetto. Per ulteriori informazioni, consulta [Ec2Resource](#).

Un'AMI personalizzata deve soddisfare i seguenti requisiti per AWS Data Pipeline poterla utilizzare correttamente per Task Runner:

- Creazione dell'AMI nella stessa regione in cui saranno eseguite le istanze. Per ulteriori informazioni, consulta [Creazione di un'AMI personalizzata](#) nella Guida per l'utente di Amazon EC2 per le istanze Linux.

- Assicurarsi che il tipo di virtualizzazione dell'AMI sia supportato dal tipo di istanza che si intende utilizzare. Ad esempio, i tipi di istanze I2 e G2 richiedono un'AMI HVM e T1, C1, M1 e i tipi di istanza M2 richiedono un'AMI PV. Per ulteriori informazioni, consulta [Tipi di virtualizzazione delle AMI Linux](#) nella Guida per l'utente di Amazon EC2 per le istanze Linux.
- Installare il seguente software:
 - Linux
 - Bash
 - wget
 - unzip
 - Java 1.6 o 1.8
 - cloud-init
- Crea e configura un utente con nome `ec2-user`.

Risoluzione dei problemi

Quando si ha un problema con AWS Data Pipeline, il sintomo più comune è che una pipeline non viene eseguita. È possibile utilizzare i dati che la console e l'interfaccia a riga di comando forniscono per individuare il problema e trovare una soluzione.

Indice

- [Individuazione di errori nelle pipeline](#)
- [Identificazione del cluster Amazon EMR che serve la tua pipeline](#)
- [Interpretazione dei dettagli sullo stato della pipeline](#)
- [Individuazione dei log di errore](#)
- [Risoluzione dei problemi più comuni](#)

Individuazione di errori nelle pipeline

La console AWS Data Pipeline è uno strumento utile che consente di monitorare visivamente lo stato delle pipeline e di individuare con facilità eventuali errori relativi a esecuzioni non riuscite o incomplete.

Individuare gli errori su esecuzioni non riuscite o incomplete con la console

1. Nella pagina List Pipelines (Elenca pipeline), se la colonna Status (Stato) di una delle istanze della pipeline mostra uno stato diverso da FINISHED (FINITO), la pipeline sta aspettando una condizione da soddisfare o ha fallito ed è necessario intervenire per risolvere il problema.
2. Nella pagina List Pipelines (Elenca pipeline), individuare la pipeline dell'istanza e selezionare il triangolo a sinistra, per espandere i dettagli.
3. Alla fine di questo pannello, scegliere View execution details (Visualizza dettagli esecuzione); si apre il pannello Instance summary (Riepilogo istanza) per mostrare i dettagli dell'istanza selezionata.
4. Nel pannello Instance summary (Riepilogo istanza), selezionare il triangolo accanto all'istanza per visualizzare ulteriori dettagli dell'istanza e scegliere Details (Dettagli), More... (Ulteriori informazioni...). Se lo stato della tua istanza selezionata è FAILED (NON RIUSCITO), la casella dettagli dispone di voci per il messaggio di errore, il `errorStackTrace` e altre informazioni. È possibile salvare le informazioni in un file. Scegli OK.

5. Nel riquadro Instance summary (Riepilogo istanze, scegliere Attempts (Tentativi) per vedere i dettagli per ogni riga di tentativo.
6. Per intervenire sull'istanza non completa o non riuscita, selezionare la casella di controllo accanto all'istanza. In questo modo vengono attivate le operazioni. Quindi, selezionare un'operazione (Rerun | Cancel | Mark Finished).

Identificazione del cluster Amazon EMR che serve la tua pipeline

Se una `EMRCluster` o `EMRActivity` fallisce e le informazioni sull'errore fornite dalla AWS Data Pipeline console non sono chiare, puoi identificare il cluster Amazon EMR che serve la tua pipeline utilizzando la console Amazon EMR. Questo ti aiuta a individuare i log forniti da Amazon EMR per ottenere maggiori dettagli sugli errori che si verificano.

Per visualizzare informazioni più dettagliate sugli errori di Amazon EMR

1. Nella console AWS Data Pipeline selezionare il triangolo accanto all'istanza della pipeline per espandere i dettagli dell'istanza.
2. Scegliere View execution details (Visualizza dettagli esecuzione) e selezionare il triangolo accanto al componente.
3. Nella colonna Details (Dettagli), scegliere More... (Altro...). Si apre la schermata di informazioni che fornisce un elenco di dettagli del componente. Individuare e copiare il valore `instanceParent` dallo schermo, ad esempio: `@EmrActivityId_xiFDD_2017-09-30T21:40:13`
4. Accedi alla console Amazon EMR, cerca un cluster con il valore `InstanceParent` corrispondente nel nome, quindi scegli Debug.

Note

Perché il pulsante Debug funzioni, la definizione della pipeline deve aver impostato l'`EmrActivityenableDebugging` opzione su `true` e l'`EmrLogUri` opzione su un percorso valido.

5. Ora che sai quale cluster Amazon EMR contiene l'errore che causa il guasto della pipeline, segui i [suggerimenti per la risoluzione dei problemi](#) nella Amazon EMR Developer Guide.

Interpretazione dei dettagli sullo stato della pipeline

I vari livelli di stato visualizzati nella console AWS Data Pipeline e nell'interfaccia a riga di comando indicano la condizione di una pipeline e dei relativi componenti. Lo stato della pipeline è semplicemente una panoramica di una pipeline; per visualizzare ulteriori informazioni, visualizzare lo stato dei singoli componenti della pipeline. È possibile eseguire questa operazione facendo clic sulla pipeline nella console o recuperando i dettagli del componente della pipeline tramite l'interfaccia a riga di comando.

Codici di stato

ACTIVATING

Il componente o la risorsa è in fase di avvio, ad esempio un'istanza EC2.

CANCELED

Il componente è stato annullato da un utente o AWS Data Pipeline prima che potesse essere eseguito. Ciò può accadere automaticamente quando si verifica un errore in un componente o in una risorsa diversa da cui dipende questo componente.

CASCADE_FAILED

Il componente o la risorsa è stato annullato a causa di un errore a cascata proveniente da una delle sue dipendenze, ma probabilmente il componente non era l'origine originale dell'errore.

DEACTIVATING

La pipeline è in fase di disattivazione.

FAILED

Il componente o la risorsa ha riscontrato un errore e ha smesso di funzionare. Quando un componente o una risorsa si guasta, è possibile che le cancellazioni e gli errori si riversino a cascata su altri componenti che ne dipendono.

FINISHED

Il componente ha completato il lavoro assegnato.

INACTIVE

La pipeline è stata disattivata.

PAUSED

Il componente è stato messo in pausa e non sta attualmente eseguendo il suo lavoro.

PENDING

La pipeline è pronta per essere attivata per la prima volta.

RUNNING

La risorsa è in esecuzione e pronta per ricevere lavoro.

SCHEDULED

L'esecuzione della risorsa è pianificata.

SHUTTING_DOWN

La risorsa si sta spegnendo dopo aver completato con successo il suo lavoro.

SKIPPED

Il componente ha saltato gli intervalli di esecuzione dopo l'attivazione della pipeline utilizzando un timestamp successivo alla pianificazione corrente.

TIMEDOUT

La risorsa ha superato la `terminateAfter` soglia ed è stata interrotta da AWS Data Pipeline. Dopo che la risorsa ha raggiunto questo stato, AWS Data Pipeline ignora i `actionOnResourceFailure` `retryTimeout` valori e e di quella risorsa. `retryDelay` Questo stato si applica solo alle risorse.

VALIDATING

La definizione della pipeline viene convalidata da AWS Data Pipeline.

WAITING_FOR_RUNNER

Il componente è in attesa che il suo cliente lavoratore recuperi un elemento di lavoro. La relazione tra componente e lavoratore cliente è controllata dai `workerGroup` campi `runsOn` o definiti da quel componente.

WAITING_ON_DEPENDENCIES

Il componente sta verificando che i prerequisiti predefiniti e configurati dall'utente siano soddisfatti prima di eseguire il suo lavoro.

Individuazione dei log di errore

Questa sezione spiega come individuare i diversi log che AWS Data Pipeline scrive, da utilizzare per stabilire l'origine di determinati guasti ed errori.

Log della pipeline

Ti consigliamo di configurare le pipeline per creare file di log in una posizione persistente, ad esempio nell'esempio seguente in cui utilizzi il `pipelineLogUri` campo sull'`Default` oggetto di una pipeline per fare in modo che tutti i componenti della pipeline utilizzino una posizione di registro Amazon S3 per impostazione predefinita (puoi ignorarlo configurando una posizione di registro in un componente specifico della pipeline).

Note

Per impostazione predefinita, Task Runner archivia i propri log in una posizione diversa, che potrebbe non essere disponibile al termine della pipeline e al termine dell'istanza che esegue Task Runner. Per ulteriori informazioni, consulta [Verifica della registrazione di Task Runner](#).

Per configurare il percorso di log utilizzando l'interfaccia a riga di comando AWS Data Pipeline in un file JSON della pipeline, iniziare il file della pipeline con il seguente testo:

```
{ "objects": [  
  {  
    "id":"Default",  
    "pipelineLogUri":"s3://mys3bucket/error_logs"  
  },  
  ...  
]
```

Dopo aver configurato una directory di log della pipeline, Task Runner crea una copia dei log nella directory, con la stessa formattazione e gli stessi nomi di file descritti nella sezione precedente sui log di Task Runner.

Step log di Hadoop Job e Amazon EMR

Con qualsiasi attività basata su Hadoop, ad esempio [HadoopActivityHiveActivity](#), [PigActivity](#) puoi visualizzare i log dei lavori di Hadoop nella posizione restituita nello slot di runtime, `hadoopJobLog` [EmrActivity](#) dispone di funzionalità di registrazione proprie e tali registri vengono archiviati utilizzando

la posizione scelta da Amazon EMR e restituiti dallo slot di runtime, . emrStepLog Per ulteriori informazioni, consulta [Visualizza i file di registro](#) nella Amazon EMR Developer Guide.

Risoluzione dei problemi più comuni

Questo argomento offre vari sintomi dei problemi di AWS Data Pipeline e i passaggi consigliati per risolverli.

Indice

- [Pipeline bloccata in stato Pending](#)
- [Componente della pipeline bloccato nello stato Waiting for Runner](#)
- [Componente della pipeline bloccato nello stato WAITING_ON_DEPENDENCIES](#)
- [L'esecuzione non inizia quando è stata programmata](#)
- [Componenti della pipeline eseguiti in ordine errato](#)
- [Il cluster EMR ha esito negativo con l'errore: "The security token included in the request is invalid" \("Il token di sicurezza incluso nella richiesta non è valido"\)](#)
- [Autorizzazioni insufficienti per accedere alle risorse](#)
- [Codice di stato: 400 Codice di errore: PipelineNotFoundException](#)
- [La creazione di una pipeline provoca un errore relativo al Security Token](#)
- [Impossibile visualizzare i dettagli della pipeline nella console](#)
- [Errore in remote runner Codice stato: 404, AWS Service: Amazon S3](#)
- [Accesso negato - Non autorizzato per eseguire la funzione datapipeline:](#)
- [Le AMI Amazon EMR meno recenti possono creare dati falsi per file CSV di grandi dimensioni](#)
- [Aumento dei limiti AWS Data Pipeline](#)

Pipeline bloccata in stato Pending

Una pipeline che sembra bloccata in stato PENDING indica che una pipeline non è stata ancora attivata o l'attivazione non è riuscita a causa di un errore nella definizione della pipeline. Verificare di non aver ricevuto eventuali errori quando è stata inviata la pipeline utilizzando l'interfaccia a riga di comando AWS Data Pipeline o durante il tentativo di salvare o attivare la pipeline tramite la console AWS Data Pipeline. Inoltre, controllare che la pipeline abbia una definizione valida.

Per visualizzare la definizione della pipeline sullo schermo utilizzando l'interfaccia a riga di comando:

```
aws datapipeline --get-pipeline-definition --pipeline-id df-EXAMPLE_PIPELINE_ID
```

Verificare che la definizione della pipeline sia completa, controllare le parentesi di chiusura e le virgole obbligatorie, verificare eventuali riferimenti mancanti e altri errori di sintassi. È consigliabile utilizzare un editor di testo in grado di convalidare visivamente la sintassi dei file JSON.

Componente della pipeline bloccato nello stato Waiting for Runner

Se la pipeline è nello stato SCHEDULED e una o più operazioni appaiono bloccate nello stato WAITING_FOR_RUNNER, verificare di aver impostato un valore valido per i campi runsOn o workerGroup per tali attività. Se entrambi i valori sono vuoti o mancanti, l'attività non può iniziare perché non vi è alcuna associazione tra l'attività e il lavoratore per eseguire l'attività. In questo caso, hai definito il lavoro ma non il computer che deve eseguire il lavoro. Se applicabile, verifica che il valore WorkerGroup assegnato al componente pipeline sia esattamente lo stesso nome e maiuscolo del valore WorkerGroup configurato per Task Runner.

Note

Se si fornisce un valore runsOn e workerGroup esiste, workerGroup verrà ignorato.

Un'altra potenziale causa di questo problema è che l'endpoint e la chiave di accesso forniti a Task Runner non sono gli stessi della AWS Data Pipeline console o del computer in cui sono installati gli strumenti AWS Data Pipeline CLI. È possibile che tu abbia creato nuove pipeline senza errori visibili, ma Task Runner esegue il sondaggio nella posizione sbagliata a causa della differenza nelle credenziali o esegue il sondaggio sulla posizione corretta con autorizzazioni insufficienti per identificare ed eseguire il lavoro specificato dalla definizione della pipeline.

Componente della pipeline bloccato nello stato WAITING_ON_DEPENDENCIES

Se la pipeline è nello stato SCHEDULED e una o più attività appaiono bloccate nello stato WAITING_ON_DEPENDENCIES, verificare che le precondizioni iniziali della pipeline siano state soddisfatte. Se le precondizioni del primo oggetto nella catena logica non sono state soddisfatte, nessuno degli oggetti che dipendono da quel primo oggetto possono essere spostati dallo stato WAITING_ON_DEPENDENCIES.

Ad esempio, considerare il seguente estratto da una definizione di pipeline. In questo caso, l'InputDataoggetto ha una precondizione «Pronto» che specifica che i dati devono esistere prima che l'InputDataoggetto sia completo. Se i dati non esistono, l'InputDataoggetto rimane nello `WAITING_ON_DEPENDENCIES` stato, in attesa che i dati specificati dal campo del percorso diventino disponibili. Tutti gli oggetti che dipendono InputData allo stesso modo da rimangono in uno `WAITING_ON_DEPENDENCIES` stato in attesa che l'InputDataoggetto raggiunga lo `FINISHED` stato.

```
{
  "id": "InputData",
  "type": "S3DataNode",
  "filePath": "s3://elasticmapreduce/samples/wordcount/wordSplitter.py",
  "schedule":{"ref":"MySchedule"},
  "precondition": "Ready"
},
{
  "id": "Ready",
  "type": "Exists"
...
}
```

Inoltre, controllare che gli oggetti abbiano le autorizzazioni richieste per accedere ai dati. Nell'esempio precedente, se le informazioni nel campo delle credenziali non disponessero delle autorizzazioni per accedere ai dati specificati nel campo del percorso, l'InputDataoggetto si bloccherebbe in uno `WAITING_ON_DEPENDENCIES` stato perché non può accedere ai dati specificati dal campo percorso, anche se tali dati esistono.

È anche possibile che a una risorsa che comunica con Amazon S3 non sia associato un indirizzo IP pubblico. Ad esempio, un `Ec2Resource` in una sottorete pubblica deve disporre di un indirizzo IP pubblico associato.

Infine, in determinate condizioni, le istanze relative a risorse possono raggiungere lo stato `WAITING_ON_DEPENDENCIES` molto prima rispetto a quando è programmato l'avvio delle attività associate, dando l'impressione che la risorsa o l'attività stia fallendo.

L'esecuzione non inizia quando è stata programmata

Controllare di aver scelto il tipo di pianificazione corretta che stabilisce se l'attività viene avviata all'inizio dell'intervallo di pianificazione (Tipo di pianificazione stile Cron) o al termine dell'intervallo di pianificazione (Tipo di pianificazione serie temporali).

Inoltre, verifica di aver specificato correttamente le date negli oggetti della pianificazione e che i `endTime` valori `startTime` and siano in formato UTC, come nell'esempio seguente:

```
{
  "id": "MySchedule",
  "startTime": "2012-11-12T19:30:00",
  "endTime": "2012-11-12T20:30:00",
  "period": "1 Hour",
  "type": "Schedule"
},
```

Componenti della pipeline eseguiti in ordine errato

È possibile notare che gli orari di inizio e di fine dei componenti della pipeline vengano eseguiti nell'ordine errato o in un'altra sequenza rispetto a quella prevista. È importante capire che l'esecuzione dei componenti della pipeline può iniziare contemporaneamente se le loro precondizioni sono soddisfatte al momento dell'avvio. In altre parole, i componenti della pipeline non vengono eseguiti in sequenza per impostazione predefinita; se è necessario un determinato ordine di esecuzione, è necessario controllare l'ordine di esecuzione con le precondizioni e i campi `dependsOn`.

Verificare che si sta utilizzando il campo `dependsOn` popolato con un riferimento ai componenti della pipeline dei prerequisiti corretti e che tutti i puntatori necessari tra componenti siano presenti per ottenere l'ordine richiesto.

Il cluster EMR ha esito negativo con l'errore: "The security token included in the request is invalid" ("Il token di sicurezza incluso nella richiesta non è valido")

Verifica i tuoi ruoli, le politiche e le relazioni di fiducia IAM come descritto in [Ruoli IAM per AWS Data Pipeline](#).

Autorizzazioni insufficienti per accedere alle risorse

Le autorizzazioni impostate sui ruoli IAM determinano se è AWS Data Pipeline possibile accedere ai cluster EMR e alle istanze EC2 per eseguire le pipeline. Inoltre, IAM fornisce il concetto di relazioni di fiducia che vanno oltre per consentire la creazione di risorse per tuo conto. Ad esempio, quando si crea una pipeline che utilizza un'istanza EC2 per eseguire un comando e spostare i dati, AWS

Data Pipeline può effettuare il provisioning di questa istanza EC2 per conto dell'utente. Se riscontri problemi, in particolare quelli che riguardano risorse a cui puoi accedere manualmente ma AWS Data Pipeline non puoi, verifica i ruoli, le politiche e le relazioni di fiducia IAM come descritto in [Ruoli IAM per AWS Data Pipeline](#).

Codice di stato: 400 Codice di errore: PipelineNotFoundException

Questo errore indica che i tuoi ruoli IAM predefiniti potrebbero non disporre delle autorizzazioni necessarie AWS Data Pipeline per funzionare correttamente. Per ulteriori informazioni, consulta [Ruoli IAM per AWS Data Pipeline](#).

La creazione di una pipeline provoca un errore relativo al Security Token

Il seguente messaggio di errore viene ricevuto quando si tenta di creare una pipeline:

Creazione della pipeline con 'pipeline_name' non riuscita. Errore: UnrecognizedClientException - Il token di sicurezza incluso nella richiesta non è valido.

Impossibile visualizzare i dettagli della pipeline nella console

Il filtro della pipeline nella console AWS Data Pipeline viene applicato alla data di inizio scheduled (programmata) per una pipeline, senza considerare quando la pipeline è stata inviata. È possibile inviare una nuova pipeline utilizzando una data di inizio programmata che si verifica nel passato, che il filtro di data predefinito potrebbe non mostrare. Per vedere i dettagli della pipeline, modificare il filtro della data per avere la garanzia che la data di inizio programmata della pipeline cada all'interno del filtro dell'intervallo di date.

Errore in remote runner Codice stato: 404, AWS Service: Amazon S3

Questo errore indica che Task Runner non è riuscito ad accedere ai tuoi file in Amazon S3. Verificare che:

- Le credenziali siano state impostate correttamente
- Il bucket Amazon S3 a cui stai tentando di accedere esiste
- Sei autorizzato ad accedere al bucket Amazon S3

Accesso negato - Non autorizzato per eseguire la funzione datapipeline:

Nei log di Task Runner, potresti visualizzare un errore simile al seguente:

- Codice stato ERRORE: 403
- Servizio AWS: DataPipeline
- Codice di errore AWS: AccessDenied
- Messaggio di errore AWS: User: arn:aws:sts: :xxxxxxxxxxx:Federated-User/I-XXXXXXXX non è autorizzato a eseguire: datapipeline:. PollForTask

Note

In questo messaggio di errore, PollForTask può essere sostituito con nomi di altre AWS Data Pipeline autorizzazioni.

Questo messaggio di errore indica che il ruolo IAM specificato richiede le autorizzazioni aggiuntive necessarie per interagire AWS Data Pipeline. Assicurati che la tua policy sui ruoli IAM contenga le seguenti righe, dove PollForTask sia sostituita dal nome dell'autorizzazione che desideri aggiungere (usa * per concedere tutte le autorizzazioni). Per ulteriori informazioni su come creare un nuovo ruolo IAM e applicare una policy ad esso, consulta [Managing IAM Policies](#) nella guida all'uso di IAM.

```
{  
  "Action": [ "datapipeline:PollForTask" ],  
  "Effect": "Allow",  
  "Resource": ["*"]  
}
```

Le AMI Amazon EMR meno recenti possono creare dati falsi per file CSV di grandi dimensioni

Sulle AMI di Amazon EMR precedenti alla 3.9 (3.8 e precedenti) AWS Data Pipeline utilizza una funzionalità personalizzata InputFormat per leggere e scrivere file CSV da utilizzare con i job. MapReduce Viene utilizzato quando il servizio organizza tavoli da e verso Amazon S3. InputFormat È stato rilevato un problema a causa del quale la lettura di record da file CSV di grandi dimensioni può comportare la produzione di tabelle che non vengono copiate correttamente. Questo problema è stato risolto nelle versioni successive di Amazon EMR. Utilizza Amazon EMR AMI 3.9 o Amazon EMR versione 4.0.0 o successiva.

Aumento dei limiti AWS Data Pipeline

Occasionalmente, è possibile superare i limiti di sistema specifici di AWS Data Pipeline. Ad esempio, il limite predefinito della pipeline è di 20 pipeline con 50 oggetti in ognuna. Se si scopre che si ha bisogno di più pipeline rispetto al limite, considerare l'unione di più pipeline per creare un numero minore di pipeline con più oggetti in ognuna. Per ulteriori informazioni sui limiti di AWS Data Pipeline, consulta [Limiti per AWS Data Pipeline](#). Tuttavia, se non è possibile risolvere il problema dei limiti tramite tecnica di unione delle pipeline, è opportuno richiedere un aumento della capacità utilizzando questo modulo: [Aumento limiti della pipeline dei dati](#).

Limiti per AWS Data Pipeline

Per garantire la capacità per tutti gli utenti, AWS Data Pipeline impone limitazioni sulle risorse da allocare e la frequenza con cui è possibile allocare tali risorse.

Indice

- [Limiti dell'account](#)
- [Limiti chiamata del servizio Web](#)
- [Considerazioni su dimensionamento](#)

Limiti dell'account

I seguenti limiti si applicano a un solo account AWS. Se hai bisogno di capacità aggiuntiva, puoi utilizzare il [modulo di richiesta del Centro Support Amazon Web Services](#) per aumentare la capacità.

Attributo	Limite	Adattabile
Numero di pipeline	100	Sì
Numero di oggetti per pipeline	100	Sì
Numero di istanze attive per oggetto	5	Sì
Numero di campi per oggetto	50	No
Numero di byte UTF8 per nome campo o identificatore	256	No
Numero di byte UTF8 per campo	10,240	No
Numero di byte UTF8 per oggetto	15.360 (inclusi i nomi campo)	No

Attributo	Limite	Adattabile
Velocità di creazione di un'istanza da un oggetto	1 ogni 5 minuti	No
Nuovi tentativi per un'attività di pipeline	5 per attività	No
Intervallo minimo tra i tentativi	2 minuti	No
Intervallo di pianificazione minimo	15 minuti	No
Numero massimo di roll-up in un singolo oggetto	32	No
Numero massimo di istanze EC2 per oggetto Ec2Resource	1	No

Limiti chiamata del servizio Web

AWS Data Pipeline limita la frequenza con cui è possibile effettuare una chiamata alle API del servizio Web. Questi limiti si applicano anche agli AWS Data Pipeline agenti che chiamano l'API del servizio Web per tuo conto, come la console, la CLI e Task Runner.

I seguenti limiti si applicano a un solo account AWS. Questo significa che l'utilizzo totale dell'account, incluso quello degli utenti, non può superare tali limiti.

La velocità di burst consente di risparmiare le chiamate del servizio Web durante i periodi di inattività e impiegarle tutte in un breve periodo di tempo. Ad esempio, CreatePipeline ha una frequenza regolare di una chiamata ogni cinque secondi. Se non chiami il servizio per 30 secondi, hai sei chiamate salvate. È quindi possibile chiamare il servizio Web sei volte in un secondo. Dal momento che questo numero è al di sotto del limite di aumento delle prestazioni e mantiene il limite di chiamate medio alla tariffa ordinaria, le chiamate non vengono limitate.

Se si supera il limite di frequenza e il limite di aumento delle prestazioni, la chiamata al servizio Web non va a buon fine e restituisce un'eccezione di throttling. L'implementazione predefinita di un worker, Task Runner, riprova automaticamente le chiamate API che falliscono con un'eccezione di limitazione. Task Runner dispone di un backoff in modo che i successivi tentativi di chiamata all'API avvengano a intervalli sempre più lunghi. Se si scrive un lavoratore, consigliamo di implementare una logica di ripetizione analoga.

Questi limiti vengono applicati rispetto a un singolo account AWS.

API	Limite frequenza regolare	Limite di aumento delle prestazioni
ActivatePipeline	1 chiamata al secondo	100 chiamate
CreatePipeline	1 chiamata al secondo	100 chiamate
DeletePipeline	1 chiamata al secondo	100 chiamate
DescribeObjects	2 chiamate al secondo	100 chiamate
DescribePipelines	1 chiamata al secondo	100 chiamate
GetPipelineDefinition	1 chiamata al secondo	100 chiamate
PollForTask	2 chiamate al secondo	100 chiamate
ListPipelines	1 chiamata al secondo	100 chiamate
PutPipelineDefinition	1 chiamata al secondo	100 chiamate
QueryObjects	2 chiamate al secondo	100 chiamate
ReportTaskProgress	10 chiamate al secondo	100 chiamate
SetTaskStatus	10 chiamate al secondo	100 chiamate
SetStatus	1 chiamata al secondo	100 chiamate
ReportTaskRunnerHeartbeat	1 chiamata al secondo	100 chiamate

API	Limite frequenza regolare	Limite di aumento delle prestazioni
ValidatePipelineDefinition	1 chiamata al secondo	100 chiamate

Considerazioni su dimensionamento

AWS Data Pipeline ridimensiona per supportare un enorme numero di attività simultanee e può essere configurato per creare automaticamente le risorse necessarie per gestire carichi di lavoro di grandi dimensioni. Queste risorse create automaticamente sono sotto il controllo dell'utente e vengono conteggiate ai fini dei limiti delle risorse dell'account AWS. Ad esempio, se configuri la creazione automatica di un cluster Amazon EMRAWS Data Pipeline a 20 nodi per elaborare i dati e il tuo AWS account ha un limite di istanze EC2 impostato su 20, potresti inavvertitamente esaurire le risorse di backfill disponibili. Di conseguenza, è necessario considerare queste limitazioni in termini di risorse nel progetto oppure aumentare i limiti dell'account in base alle necessità.

Se hai bisogno di capacità aggiuntiva, puoi utilizzare il [modulo di richiesta del Centro Support Amazon Web Services](#) per aumentare la capacità.

Risorse di AWS Data Pipeline

Le seguenti sono risorse che ti aiutano a utilizzare AWS Data Pipeline.

- [AWS Data Pipeline Informazioni sul prodotto](#): pagina Web principale che include le informazioni su AWS Data Pipeline.
- [AWS Data Pipeline Domande frequenti tecniche](#): copre le 20 domande principali poste dagli sviluppatori su questo prodotto.
- [Note di rilascio](#): forniscono una panoramica di alto livello della versione corrente. In particolare, vengono indicate tutte le nuove funzioni, le correzioni e tutti i problemi noti.
- Forum di [discussione AWS Data Pipeline: forum](#) basato su una community per sviluppatori dedicato alla discussione di questioni tecniche correlate ad Amazon Web Services.
- Corsi [e seminari](#): collegamenti a corsi basati su ruoli e di specializzazione nonché a corsi gestiti dall'utente per affinare le proprie competenze e acquisire esperienza pratica.
- [AWS Centro per sviluppatori](#): esplora i tutorial, scarica gli strumenti e scopri di più sugli eventi per gli AWS sviluppatori.
- [AWS Strumenti per sviluppatori](#): collegamenti a strumenti per sviluppatori, kit SDK, kit di strumenti IDE e strumenti a riga di comando per lo sviluppo e la gestione delle AWS applicazioni.
- [Centro risorse per iniziare](#): scopri come configurare il tuo Account AWS, entrare a far parte della AWS community e lanciare la tua prima applicazione.
- Esercitazioni [pratiche: scopri le esercitazioni](#) pratiche: step-by-step scopri le esercitazioni pratiche per avviare la tua prima applicazione su AWS.
- [AWS Whitepaper](#): collegamenti a un elenco completo di AWS whitepaper tecnici, relativi ad argomenti come l'architettura, la sicurezza e l'economia, creati da AWS Solution Architect o da altri esperti tecnici.
- [AWS Support Centro](#): il centro in cui creare e gestire i tuoi casi AWS Support. Include inoltre link ad altre risorse utili, quali forum, domande frequenti di tipo tecnico, stato d'integrità del servizio e AWS Trusted Advisor.
- [AWS Support](#): pagina Web principale che include le informazioni su AWS Support one-on-one, un canale di assistenza rapida che aiuta a creare ed eseguire applicazioni nel cloud.
- [Contatti](#) - Un punto di contatto centrale per richieste relative a fatturazione, account, eventi, uso illecito e altre questioni relative ad AWS.

- [AWS Termini di utilizzo del sito](#): informazioni dettagliate sul copyright e i marchi, l'account, la licenza, l'accesso al sito e altri argomenti.

Cronologia dei documenti

Questa documentazione è associata alla versione 2012-10-29 di AWS Data Pipeline

Modifica	Descrizione	Data di rilascio
È stata aggiunta la documentazione per l'esecuzione di determinate procedure utilizzando la AWS CLI. Procedure relative alla AWS Data Pipeline console rimosse.	Per ulteriori informazioni, consulta Clonazione della pipeline , Visualizza log pipeline e Crea una pipeline dai modelli di Data Pipeline utilizzando la CLI .	26 maggio 2023
Sono stati aggiunti altri contenuti ed esempi per la migrazione AWS Data Pipeline da altri servizi alternativi.	È stato aggiornato l'argomento per la migrazione AWS Data Pipeline a AWS Step Functions o Amazon MWAA con ulteriori informazioni su ciascuna alternativa, mappature concettuali tra i servizi ed esempi. AWS Glue Per ulteriori informazioni, consulta Migrazione dei carichi di lavoro da AWS Data Pipeline .	31 marzo 2023
Sono state aggiunte informazioni sul supporto di IMDSv2 di AWS Data Pipeline.	AWS Data Pipeline supporta IMDSv2 per le risorse Amazon EMR e Amazon EC2. Per ulteriori informazioni, consulta Protezione dei dati in AWS Data Pipeline , EmrCluster e Ec2Resource .	16 dicembre 2022
È stato aggiunto un argomento per la migrazione AWS Data Pipeline da altri servizi alternativi.	Ora esistono altri AWS servizi che offrono ai clienti una migliore esperienza di integrazione dei dati. Puoi migrare i casi d'uso tipici AWS Data Pipeline su AWS Step Functions o Amazon MWAA. AWS Glue Per ulteriori informazioni, consulta Migrazione dei carichi di lavoro da AWS Data Pipeline .	16 dicembre 2022

Modifica	Descrizione	Data di rilascio
<p>Aggiornati gli elenchi delle istanze Amazon EC2 e Amazon EMR supportate.</p> <p>Aggiornato l'elenco di ID delle AMI HVM (Hardware Virtual Machine) utilizzate per le istanze.</p>	<p>Aggiornati gli elenchi delle istanze Amazon EC2 e Amazon EMR supportate. Per ulteriori informazioni, consulta Tipi di istanza supportati per attività di lavoro delle pipeline.</p> <p>Aggiornato l'elenco di ID delle AMI HVM (Hardware Virtual Machine) utilizzate per le istanze. Per ulteriori informazioni, consulta Sintassi e cerca <code>imageId</code>.</p>	<p>9 novembre 2018</p>
<p>È stata aggiunta una configurazione per collegare i volumi Amazon EBS ai nodi del cluster e per avviare un cluster Amazon EMR in una sottorete privata.</p>	<p>Aggiunte le opzioni di configurazione a un oggetto <code>EMRCluster</code>. Puoi utilizzare queste opzioni nelle pipeline che utilizzano cluster Amazon EMR.</p> <p>Utilizza i <code>TaskEbsConfiguration</code> campi <code>coreEbsConfiguration</code> <code>masterEbsConfiguration</code>, e per configurare il collegamento dei volumi Amazon EBS ai nodi core, master e task nel cluster Amazon EMR. Per ulteriori informazioni, consulta Collegare i volumi EBS ai nodi del cluster.</p> <p>Utilizza i <code>ServiceAccessSecurityGroup</code> di campi <code>emrManagedMasterSecurityGroupId</code> <code>emrManagedSlaveSecurityGroupId</code>, e per configurare un cluster Amazon EMR in una sottorete privata. Per ulteriori informazioni, consulta Configurazione di un cluster Amazon EMR in una sottorete privata.</p> <p>Per ulteriori informazioni sulla sintassi <code>EMRCluster</code>, consulta EmrCluster.</p>	<p>19 aprile 2018</p>

Modifica	Descrizione	Data di rilascio
È stato aggiunto l'elenco delle istanze Amazon EC2 e Amazon EMR supportate.	Aggiunto l'elenco di istanze che AWS Data Pipeline crea per impostazione predefinita, se non si specifica un tipo di istanza nella definizione della pipeline. È stato aggiunto un elenco di istanze Amazon EC2 e Amazon EMR supportate. Per ulteriori informazioni, consulta Tipi di istanza supportati per attività di lavoro delle pipeline .	22 marzo 2018
Aggiunto il supporto per pipeline on demand.	<ul style="list-style-type: none"> • Aggiunto il supporto per pipeline on demand, che consente di eseguire nuovamente una pipeline attivandola di nuovo. 	22 febbraio 2016
Supporto aggiuntivo per database RDS	<ul style="list-style-type: none"> • Aggiunti <code>rdsInstanceId</code>, <code>region</code> e <code>jdbcDriverJarUri</code> a RdsDatabase. • Aggiornato database in SqlActivity per supportare anche <code>RdsDatabase</code>. 	17 agosto 2015
Ulteriore supporto di JDBC	<ul style="list-style-type: none"> • Aggiornato database in SqlActivity per supportare anche <code>JdbcDatabase</code>. • È stato aggiunto <code>jdbcDriverJarUri</code> a JdbcDatabase. • Aggiunto <code>initTimeout</code> a Ec2Resource e EmrCluster. • È stato aggiunto <code>runAsUser</code> a Ec2Resource. 	7 luglio 2015

Modifica	Descrizione	Data di rilascio
HadoopActivity, Zona di disponibilità e supporto Spot	<ul style="list-style-type: none"> • Aggiunto il supporto per l'invio di lavoro parallelo Al cluster Hadoop. Per ulteriori informazioni, consulta HadoopActivity. • Aggiunta la possibilità di richiedere le istanze Spot con Ec2Resource e EmrCluster. • Aggiunta la possibilità di avviare risorse EmrCluster in una determinata zona di disponibilità. 	1 giugno 2015
Disattivazione pipeline	Aggiunto il supporto per la disattivazione delle pipeline attive. Per ulteriori informazioni, consulta Disattivazione pipeline .	7 aprile 2015
Aggiornati modelli e console	Aggiunti nuovi modelli. È stato aggiornato il capitolo Guida introduttiva per utilizzare il ShellCommandActivity modello Guida introduttiva. Per ulteriori informazioni, consulta Crea una pipeline dai modelli di Data Pipeline utilizzando la CLI .	25 novembre 2014
Supporto per VPC	Aggiunto il supporto per avviare le risorse in un cloud privato virtuale (VPC).	12 marzo 2014
Supporto di una regione	Aggiunto il supporto per diverse regioni del servizio. Oltre a us-east-1 , AWS Data Pipeline è supportato in eu-west-1 , ap-northeast-1 , ap-southeast-2 e us-west-2 .	20 febbraio 2014

Modifica	Descrizione	Data di rilascio
Supporto di Amazon Redshift	È stato aggiunto il supporto per Amazon Redshift in AWS Data Pipeline, incluso un nuovo modello di console (Copy to Redshift) e un tutorial per dimostrare il modello. Per ulteriori informazioni, vedere Copia i dati su Amazon Redshift utilizzando AWS Data Pipeline , RedshiftDataNode , RedshiftDatabase e RedshiftCopyActivity .	6 novembre 2013
PigActivity	Aggiunto PigActivity, che fornisce supporto nativo per Pig. Per ulteriori informazioni, consulta PigActivity .	15 ottobre 2013
Nuovo modello di console, attività e formato di dati	È stato aggiunto il nuovo modello di console CrossRegion DynamoDB Copy, incluso il nuovo HiveCopyActivity e DynamoDB.ExportDataFormat	21 agosto 2013
Guasti di una delle dipendenze e riesecuzioni	Aggiunte informazioni sul comportamento di guasti di una delle dipendenze e di riesecuzioni di AWS Data Pipeline. Per ulteriori informazioni, consulta Guasti di una delle dipendenze e riesecuzioni .	8 agosto 2013
Risoluzione dei problemi video	Aggiunto il video di base di risoluzione dei problemi di AWS Data Pipeline. Per ulteriori informazioni, consulta Risoluzione dei problemi .	17 luglio 2013
Modifica delle pipeline attive	Aggiunte ulteriori informazioni sulla modifica delle pipeline attive e sulla riesecuzione dei componenti della pipeline. Per ulteriori informazioni, consulta Modifica della pipeline .	17 luglio 2013
Utilizza le risorse in regioni diverse	Aggiunte ulteriori informazioni sull'utilizzo delle risorse in regioni diverse. Per ulteriori informazioni, consulta Utilizzo di una pipeline con risorse in più regioni .	17 giugno 2013
Stato WAITING_ON_DEPENDENCIES	Stato CHECKING_PRECONDITIONS modificato in WAITING_ON_DEPENDENCIES e aggiunto il campo runtime @waitingOn per gli oggetti delle pipeline.	20 maggio 2013

Modifica	Descrizione	Data di rilascio
Dinamo DB DataFormat	È stato aggiunto un modello DynamoDB. DataFormat	23 aprile 2013
Video log Web dei processi e supporto delle istanze Spot	Ha presentato il video «Elabora i registri Web con AWS Data Pipeline, Amazon EMR e Hive» e il supporto per le istanze Spot di Amazon EC2.	21 febbraio 2013
	Versione iniziale della Guida per sviluppatori di AWS Data Pipeline.	20 dicembre 2012

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.