Guida per l'utente

AWS HealthOmics

Version latest

AWS HealthOmics: Guida per l’utente

Table of Contents

Che cos'è AWS HealthOmics?

AWS HealthOmics è un servizio idoneo all'HIPAA che accelera i test diagnostici clinici, la scoperta di farmaci e la ricerca agricola gestendo completamente la complessa infrastruttura alla base dei flussi di lavoro bioinformatici. HealthOmics supporta i linguaggi di flusso di lavoro standard del settore (WDL, Nextflow, CWL) e ridimensiona perfettamente l'infrastruttura bioinformatica per supportare i dati di decine di migliaia di test al giorno, il tutto con un costo per campione prevedibile. HealthOmics gestisce le complessità tecniche come la gestione delle risorse di elaborazione e la manutenzione dei motori del flusso di lavoro in modo da poterti concentrare interamente sulle scoperte scientifiche.

Argomenti

Avviso importante

HealthOmics è destinato esclusivamente al trasferimento, all'archiviazione, alla formattazione o alla visualizzazione di dati e alla fornitura di supporto infrastrutturale e di configurazione per la gestione dei flussi di lavoro. HealthOmics non sostituisce la consulenza, la diagnosi o il trattamento medico professionale e non è destinato a curare, trattare, mitigare, prevenire o diagnosticare alcuna malattia o condizione di salute. L'utente è responsabile dell'istituzione della revisione umana nell'ambito di qualsiasi utilizzo AWS HealthOmics, anche in associazione a, di qualsiasi prodotto di terze parti destinato a informare il processo decisionale clinico.

HealthOmics features

Casi d'uso primari per: HealthOmics

- Diagnostica clinica: crea e ridimensiona i flussi di lavoro dei test diagnostici con costi prevedibili e un'infrastruttura completamente gestita che cresce con il volume dei test.

- Scoperta di farmaci: accelera la ricerca terapeutica orchestrando modelli di base biologici su larga scala, che consentono una rapida iterazione tra milioni di potenziali candidati.
- Ricerca agricola: migliora le caratteristiche delle colture come la tolleranza alla siccità e la resistenza ai parassiti attraverso flussi di lavoro basati sull'intelligenza artificiale che migliorano la sicurezza alimentare e la produttività agricola.

HealthOmicsPrincipali vantaggi di:

- Scalabilità: scalabilità dei flussi di lavoro su oltre 100.000 flussi di lavoro simultanei CPUs per supportare decine di migliaia di test al giorno con una gestione dell'infrastruttura pari a zero e un costo per campione prevedibile.
- Concentrati sulla scienza, non sull'infrastruttura: utilizza linguaggi di flusso di lavoro familiari e gestisci AWS automaticamente l'orchestrazione dell'infrastruttura e APIs la gestione dei dati dietro le quinte.
- Mantieni la conformità: audit trail completi, tracciamento della provenienza dei dati e infrastruttura conforme all'HIPAA progettata per i flussi di lavoro clinici supportano lo sviluppo di soluzioni che soddisfano i requisiti out-of-the-box normativi.

HealthOmics è costituito da tre componenti principali:

- HealthOmics flussi di lavoro: esegui calcoli bioinformatici su un'infrastruttura scalabile e con provisioning automatico.
- HealthOmics archiviazione: archivia e condividi petabyte di dati genomici in modo efficiente a un basso costo per gigabase.
- HealthOmics analisi: prepara i dati genomici per analisi multiomiche e multimodali.

Utilizzate questi componenti in modo indipendente o combinateli per ottenere una soluzione. end-to-end

HealthOmics concetti

Questo argomento descrive le definizioni dei concetti e dei termini chiave specifici di HealthOmics, per aiutarti a comprendere la terminologia HealthOmics utilizzata in questa guida.

Argomenti

Flussi di lavoro

Con HealthOmics Workflows, puoi elaborare e analizzare i tuoi dati genomici.

- Flusso di lavoro: la definizione generale di un processo end-to-end, inclusi parametri e riferimenti agli strumenti. Le definizioni del flusso di lavoro possono essere espresse come WDL, Nextflow o CWL. Ogni flusso di lavoro creato ha un identificatore univoco.
- Esegui: una singola chiamata di un flusso di lavoro. Una singola esecuzione utilizza i dati di input definiti e produce un output. Ogni esecuzione creata ha un identificatore univoco.
- Attività: i singoli processi all'interno di una corsa. HealthOmics I flussi di lavoro utilizzano queste specifiche di elaborazione definite per eseguire l'attività. Ogni attività ha un identificatore univoco.
- Gruppo di esecuzione: un gruppo di esecuzioni per il quale è possibile impostare la vCPU massima, la durata massima o il numero massimo di esecuzioni simultanee per limitare le risorse di calcolo utilizzate per esecuzione. È possibile specificare e configurare le priorità per le esecuzioni all'interno di un gruppo di esecuzione. Ad esempio, è possibile specificare che venga eseguita un'esecuzione ad alta priorità prima di un'esecuzione con priorità inferiore, creando una coda prioritaria. L'utilizzo di un Run Group è facoltativo e ogni Run Group ha un identificatore univoco.

Storage

L'archiviazione dei dati è suddivisa in archivi di sequenze, per le sequenze genomiche e le informazioni correlate, e un archivio di riferimento, per tutti i genomi di riferimento. I termini seguenti descrivono le implementazioni specifiche di. HealthOmics

- Sequence store: un archivio dati per l'archiviazione di file genomici. All'interno è possibile avere uno o più archivi di sequenze. HealthOmics Le autorizzazioni di accesso e AWS KMS la crittografia possono essere impostate su un archivio di sequenze per controllare chi ha accesso ai dati.
- Set di lettura: un set di lettura è un'astrazione delle letture genomiche, archiviate nei formati FASTQ, BAM o CRAM. I set di lettura possono essere importati negli archivi di sequenze e annotati con metadati. È possibile applicare le autorizzazioni ai set di lettura utilizzando il controllo degli accessi basato sugli attributi (ABAC).

- Riferimento: un riferimento al genoma viene utilizzato con le letture per identificare dove in un genoma viene mappata una lettura specifica o un gruppo di letture. Questi sono in formato FASTA e memorizzati nell'archivio di riferimento.
- Archivio di riferimento: un archivio dati per l'archiviazione dei genomi di riferimento. Puoi avere un unico archivio di riferimento in ogni account e regione.

Analisi

Puoi trasformare e analizzare i tuoi dati genomici con HealthOmics Analytics. Crea un negozio di varianti o un archivio di annotazioni per includere informazioni aggiuntive per le tue query.

- Archivio varianti: archivio dati che archivia i dati delle varianti su scala di popolazione. Gli archivi di varianti supportano sia gli input genomici Variant Call Format (gvCF) che gli input VCF.
- Archivio di annotazioni: un archivio dati che rappresenta un database di annotazioni, ad esempio uno contenuto in un file TSV/CSV, VCF o General Feature Format (). GFF3 Gli archivi di annotazioni vengono mappati sullo stesso sistema di coordinate degli archivi di varianti durante l'importazione.

Servizi correlati

I seguenti servizi funzionano con. HealthOmics

- Amazon Elastic Container Registry: ogni flusso di lavoro privato utilizza un'immagine Amazon ECR (in un repository Amazon ECR privato) per contenere tutti i file eseguibili, le librerie e gli script necessari per eseguire il flusso di lavoro.
- Amazon Simple Storage Service: Amazon S3 fornisce lo storage di file per i dati di Store e Workflow.
- AWS Lake Formation — Lake Formation gestisce l'accesso ai dati ai tuoi archivi di dati Analytics.
- Amazon Athena: usa Athena per eseguire query sui tuoi negozi Variant.
- Amazon SageMaker AI: utilizza l' SageMaker intelligenza artificiale per eseguire HealthOmics attività utilizzando i notebook Jupyter.
- GitHub connections— Usa le connessioni per connettere i tuoi repository di codice esterni ai tuoi flussi di lavoro. HealthOmics

Come accedere HealthOmics

Puoi accedere alle AWS HealthOmics funzionalità utilizzando la console di gestione, la CLI SDKs o l'API.

- AWS Console di gestione: fornisce un'interfaccia Web che è possibile utilizzare per accedere HealthOmics.
- AWS Command Line Interface (AWS CLI) — Fornisce comandi per un'ampia gamma di AWS servizi AWS HealthOmics, inclusi ed è supportato su Windows, macOS e Linux. Per ulteriori informazioni sull'installazione di AWS CLI, vedere AWS Command Line Interface.
- AWS SDKs — AWS fornisce SDKs (Software Development Kit) costituiti da librerie e codice di esempio per vari linguaggi e piattaforme di programmazione (inclusi Java, Python, Ruby, .NET, iOS e Android). SDKs Forniscono un modo comodo per l'uso a livello di codice. HealthOmics Per ulteriori informazioni, consulta l'AWS SDK Developer Center.
- AWS API: puoi utilizzare le operazioni API per accedere e gestire in modo HealthOmics programmatico. Per ulteriori informazioni, consulta la Documentazione di riferimento delle API di HealthOmics .

Regioni ed endpoint per AWS HealthOmics

Per un elenco completo delle regioni e degli endpoint, consulta la Guida AWS generale.

Oltre alle AWS regioni che sono attive per impostazione predefinita, ci sono anche regioni Opt-in che devono essere attivate. Per ulteriori informazioni su come attivare o disattivare una regione, consulta Specificare le AWS regioni che il tuo account può utilizzare nella guida alla gestione dell' AWS account.

Ulteriori informazioni

Scopri di più grazie HealthOmics a questi workshop e tutorial:

- HealthOmics workshop: workshop dall'HealthOmics inizio alla fine
- AWS risorse genomiche: archivi Amazon ECR pubblici relativi alla genomica
- Tutorial in Python: tutorial per notebook Jupyter su storage, analisi e flussi di lavoro GitHub HealthOmics

Acquisisci HealthOmics familiarità con AWS strumenti aggiuntivi che forniscono:

- Linter WDL — HealthOmics linter per WDL
- Linter Nextflow — linter per Nextflow HealthOmics
- HealthOmics Strumento di supporto Amazon ECR — Strumento di supporto Amazon ECR per HealthOmics
- HealthOmics tools on GitHub : strumenti con cui lavorare HealthOmics (Transfer manager, URI parser, Omics rerun, Run analyzer).

AWS HealthOmics modifica della disponibilità dell'archivio delle varianti e dell'archivio delle annotazioni

Dopo un'attenta valutazione, abbiamo deciso di chiudere i negozi di AWS HealthOmics varianti e i negozi di annotazioni a nuovi clienti a partire dal 7 novembre 2025. I clienti esistenti possono continuare a utilizzare il servizio normalmente.

La sezione seguente descrive le opzioni di migrazione per aiutarti a spostare i tuoi negozi di varianti e i negozi di analisi verso nuove soluzioni. Per qualsiasi domanda o dubbio, crea una richiesta di supporto all'indirizzo support.console.aws.amazon.com.

Argomenti

Panoramica delle opzioni di migrazione

Le seguenti opzioni di migrazione offrono un'alternativa all'utilizzo degli archivi di varianti e degli archivi di annotazioni:

1. Utilizzate l'implementazione HealthOmics di riferimento della logica ETL fornita.

   Usa i bucket di tabella S3 per l'archiviazione e continua a utilizzare i servizi di analisi esistenti. AWS
2. Crea una soluzione utilizzando una combinazione di servizi esistenti AWS .

   Per ETL, puoi scrivere lavori Glue ETL personalizzati o utilizzare codice HAIL o GLOW open source su EMR per trasformare i dati delle varianti.

   Usa i table bucket S3 per l'archiviazione e continua a utilizzare i servizi di analisi esistenti AWS
3. Seleziona un AWS partner che offra una variante e un'alternativa all'annotation store.

Opzioni di migrazione per la logica ETL

Considerate le seguenti opzioni di migrazione per la logica ETL:

1. HealthOmics fornisce l'attuale logica ETL del negozio di varianti come flusso di lavoro di riferimento. HealthOmics È possibile utilizzare il motore di questo flusso di lavoro per alimentare esattamente lo stesso processo ETL dei dati delle varianti dell'archivio delle varianti, ma con il pieno controllo sulla logica ETL.

   Questo flusso di lavoro di riferimento è disponibile su richiesta. Per richiedere l'accesso, crea una richiesta di supporto all'indirizzo support.console.aws.amazon.com.
2. Per trasformare i dati delle varianti, puoi scrivere lavori Glue ETL personalizzati o utilizzare codice HAIL o GLOW open source su EMR.

Opzioni di migrazione per lo storage

In sostituzione dell'archivio dati ospitato nel servizio, puoi utilizzare i bucket di tabella Amazon S3 per definire uno schema di tabella personalizzato. Per ulteriori informazioni sui table bucket, consulta Table bucket nella Amazon S3 User Guide.

Puoi usare i table bucket per tabelle Iceberg completamente gestite in Amazon S3.

Puoi presentare una richiesta di supporto per richiedere al HealthOmics team di migrare i dati dalla tua variante o dall'archivio di annotazioni al bucket di tabelle Amazon S3 che hai configurato.

Dopo aver inserito i dati nel bucket di tabelle Amazon S3, puoi eliminare gli archivi di varianti e gli archivi di annotazioni. Per ulteriori informazioni, consulta Eliminazione degli archivi di analisi. HealthOmics

Analisi

Per l'analisi dei dati, continua a utilizzare servizi di AWS analisi come Amazon Athena, Amazon EMR, Amazon Redshift o Amazon Quick Suite.

AWS Partner

Puoi collaborare con un AWS partner che fornisce ETL personalizzabili, schemi di tabelle, strumenti di query e analisi integrati e interfacce utente per l'interazione con i dati.

Esempi

Gli esempi seguenti mostrano come creare tabelle adatte alla memorizzazione di dati VCF e GVCF.

Athena DDL

È possibile utilizzare il seguente esempio DDL in Athena per creare una tabella adatta alla memorizzazione di dati VCF e GVCF in un'unica tabella. Questo esempio non è l'esatto equivalente della struttura dell'archivio delle varianti, ma funziona bene per un caso d'uso generico.

Crea i tuoi valori per DATABASE_NAME e TABLE_NAME quando crei la tabella.

```
 CREATE TABLE <DATABASE_NAME>. <TABLE_NAME> (
  sample_name string,
  variant_name string COMMENT 'The ID field in VCF files, '.' indicates no name',
  chrom string,
  pos bigint,
  ref string,
  alt array <string>,
  qual double,
  filter string,
  genotype string,
  info map <string, string>,
  attributes map <string, string>,
  is_reference_block boolean COMMENT 'Used in GVCF for non-variant sites')
PARTITIONED BY (bucket(128, sample_name), chrom)
LOCATION '{URL}/'
TBLPROPERTIES (
  'table_type'='iceberg',
  'write_compression'='zstd'
);
```

Crea tabelle usando Python (senza Athena)

Il seguente esempio di codice Python mostra come creare le tabelle senza usare Athena.

```
 import boto3
from pyiceberg.catalog import Catalog, load_catalog
from pyiceberg.schema import Schema
from pyiceberg.table import Table
from pyiceberg.table.sorting import SortOrder, SortField, SortDirection, NullOrder
```

```
from pyiceberg.partitioning import PartitionSpec, PartitionField
from pyiceberg.transforms import IdentityTransform, BucketTransform
from pyiceberg.types import (
    NestedField,
    StringType,
    LongType,
    DoubleType,
    MapType,
    BooleanType,
    ListType
)


def load_s3_tables_catalog(bucket_arn: str) -> Catalog:
    session = boto3.session.Session()
    region = session.region_name or 'us-east-1'

    catalog_config = {
        "type": "rest",
        "warehouse": bucket_arn,
        "uri": f"https://s3tables.{region}.amazonaws.com/iceberg",
        "rest.sigv4-enabled": "true",
        "rest.signing-name": "s3tables",
        "rest.signing-region": region
    }

    return load_catalog("s3tables", **catalog_config)


def create_namespace(catalog: Catalog, namespace: str) -> None:
    try:
        catalog.create_namespace(namespace)
        print(f"Created namespace: {namespace}")
    except Exception as e:
        if "already exists" in str(e):
            print(f"Namespace {namespace} already exists.")
        else:
            raise e


def create_table(catalog: Catalog, namespace: str, table_name: str, schema: Schema,
                 partition_spec: PartitionSpec = None, sort_order: SortOrder = None) ->
 Table:
    if catalog.table_exists(f"{namespace}.{table_name}"):
```

```
        print(f"Table {namespace}.{table_name} already exists.")
        return catalog.load_table(f"{namespace}.{table_name}")

    create_table_args = {
        "identifier": f"{namespace}.{table_name}",
        "schema": schema,
        "properties": {"format-version": "2"}
    }

    if partition_spec is not None:
        create_table_args["partition_spec"] = partition_spec
    if sort_order is not None:
        create_table_args["sort_order"] = sort_order

    table = catalog.create_table(**create_table_args)
    print(f"Created table: {namespace}.{table_name}")
    return table


def main(bucket_arn: str, namespace: str, table_name: str):
    # Schema definition
    genomic_variants_schema = Schema(
        NestedField(1, "sample_name", StringType(), required=True),
        NestedField(2, "variant_name", StringType(), required=True),
        NestedField(3, "chrom", StringType(), required=True),
        NestedField(4, "pos", LongType(), required=True),
        NestedField(5, "ref", StringType(), required=True),
        NestedField(6, "alt", ListType(element_id=1000, element_type=StringType(),
 element_required=True), required=True),
        NestedField(7, "qual", DoubleType()),
        NestedField(8, "filter", StringType()),
        NestedField(9, "genotype", StringType()),
        NestedField(10, "info", MapType(key_type=StringType(), key_id=1001,
 value_type=StringType(), value_id=1002)),
        NestedField(11, "attributes", MapType(key_type=StringType(), key_id=2001,
 value_type=StringType(), value_id=2002)),
        NestedField(12, "is_reference_block", BooleanType()),
        identifier_field_ids=[1, 2, 3, 4]
    )

    # Partition and sort specifications
    partition_spec = PartitionSpec(
        PartitionField(source_id=1, field_id=1001, transform=BucketTransform(128),
 name="sample_bucket"),
```

```
        PartitionField(source_id=3, field_id=1002, transform=IdentityTransform(),
 name="chrom")
    )

    sort_order = SortOrder(
        SortField(source_id=3, transform=IdentityTransform(),
 direction=SortDirection.ASC, null_order=NullOrder.NULLS_LAST),
        SortField(source_id=4, transform=IdentityTransform(),
 direction=SortDirection.ASC, null_order=NullOrder.NULLS_LAST)
    )

    # Connect to catalog and create table
    catalog = load_s3_tables_catalog(bucket_arn)
    create_namespace(catalog, namespace)
    table = create_table(catalog, namespace, table_name, genomic_variants_schema,
 partition_spec, sort_order)

    return table


if __name__ == "__main__":
    bucket_arn = 'arn:aws:s3tables:<REGION>:<ACCOUNT_ID>:bucket/<TABLE_BUCKET_NAME'
    namespace = "variant_db"
    table_name = "genomic_variants"

    main(bucket_arn, namespace, table_name)
```

Configurazione HealthOmics

Per configurare AWS HealthOmics, iscriviti a un utente Account AWS, crea un utente amministrativo e gestisci in modo sicuro l'accesso per altri utenti.

Argomenti

Registrati per un Account AWS

Se non ne hai uno Account AWS, completa i seguenti passaggi per crearne uno.

Per iscriverti a un Account AWS

1. Apri la https://portal.aws.amazon.com/billing/registrazione.
2. Segui le istruzioni online.

   Nel corso della procedura di registrazione riceverai una telefonata o un messaggio di testo e ti verrà chiesto di inserire un codice di verifica attraverso la tastiera del telefono.

   Quando ti iscrivi a un Account AWS, Utente root dell'account AWSviene creato un. 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 e utilizza solo l'utente root per eseguire attività che richiedono l'accesso di un utente root.

AWS ti invia un'email di conferma dopo il completamento della procedura di registrazione. In qualsiasi momento, puoi visualizzare l'attività corrente del tuo account e gestirlo accedendo a https://aws.amazon.com/e scegliendo Il mio account.

Crea un utente con accesso amministrativo

Dopo esserti registrato Account AWS, proteggi Utente root dell'account AWS AWS IAM Identity Center, abilita e crea un utente amministrativo in modo da non utilizzare l'utente root per le attività quotidiane.

Proteggi i tuoi Utente root dell'account AWS

1. Accedi Console di gestione AWScome proprietario dell'account scegliendo Utente root e inserendo il tuo indirizzo Account AWS email. Nella pagina successiva, inserisci la password.

   Per informazioni sull'accesso utilizzando un utente root, consulta la pagina Signing in as the root user della Guida per l'utente di Accedi ad AWS .
2. Abilita l'autenticazione a più fattori (MFA) per l'utente root.

   Per istruzioni, consulta Abilitare un dispositivo MFA virtuale per l'utente Account AWS root (console) nella Guida per l'utente IAM.

Crea un utente con accesso amministrativo

1. Abilita Centro identità IAM.

   Per istruzioni, consulta Abilitazione di AWS IAM Identity Center nella Guida per l'utente di AWS IAM Identity Center .
2. In IAM Identity Center, assegna l'accesso amministrativo a un utente.

   Per un tutorial sull'utilizzo di IAM Identity Center directory come fonte di identità, consulta Configurare l'accesso utente con l'impostazione predefinita 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 AWS Accedere al portale di accesso nella Guida per l'Accedi ad AWS utente.

Assegna l'accesso a ulteriori utenti

1. In IAM Identity Center, crea un set di autorizzazioni conforme alla best practice dell'applicazione di autorizzazioni con il privilegio minimo.

   Segui le istruzioni riportate nella pagina Creazione di un set di autorizzazioni nella Guida per l'utente di AWS IAM Identity Center .
2. Assegna al gruppo prima gli utenti e poi l'accesso con autenticazione unica (Single Sign-On).

   Per istruzioni, consulta Aggiungere gruppi nella Guida per l'utente di AWS IAM Identity Center .

Crea autorizzazioni IAM per HealthOmics

Per utilizzarle HealthOmics, configura le seguenti autorizzazioni IAM:

- Politiche basate sull'identità IAM a cui possono accedere gli utenti del tuo account. HealthOmics
- Un ruolo di servizio IAM per accedere HealthOmics alle risorse per tuo conto.
- Autorizzazioni in altri servizi (come Lake Formation e Amazon ECR) per consentire agli utenti e al HealthOmics servizio di accedere alle risorse.

Per ulteriori informazioni sulla configurazione delle autorizzazioni IAM per, consulta. HealthOmics Autorizzazioni IAM per HealthOmics

Connect con repository di codice esterni

Con AWS HealthOmics, puoi gestire i tuoi flussi di lavoro utilizzando repository basati su Git tramite. AWS CodeConnections HealthOmics utilizza questa connessione per accedere ai repository del codice sorgente.

Prima di utilizzare archivi di codice esterni, segui la guida alla configurazione delle connessioni per iniziare a utilizzarli. AWS CodeConnections Verifica di aver creato le politiche e le autorizzazioni IAM appropriate per il tuo AWS account. Per un elenco dei provider Git supportati e ulteriori informazioni, vedi Per quali provider di terze parti posso creare connessioni? .

Creare una connessione

Per creare una connessione con il tuo provider di repository preferito, segui il tutorial Crea una connessione.

Utilizzo dell'interfaccia a riga di comando di Amazon Q con HealthOmics

L'interfaccia a riga di comando di Amazon Q fornisce interazioni in linguaggio naturale AWS HealthOmics, che consentono di eseguire flussi di lavoro genomici complessi e attività di analisi utilizzando comandi conversazionali. Per utilizzare Amazon Q CLI, assicurati di configurare le autorizzazioni IAM per HealthOmics e altri servizi (come Amazon ECR o CloudWatch Amazon S3) per consentire ad Amazon Q di accedere alle loro risorse.

Il tutorial sull'intelligenza artificiale generativa di HealthOmics Agentic fornisce una step-by-step guida per configurare i file di contesto e consentire alla CLI di Amazon Q di creare, eseguire e ottimizzare i flussi di lavoro. AWS HealthOmics

Guida introduttiva con HealthOmics

Per iniziare HealthOmics, assicurati di aver configurato correttamente le autorizzazioni e i ruoli IAM per HealthOmics.

Utilizzo di un flusso di lavoro Ready2Run nella console HealthOmics

L'esercizio seguente mostra come utilizzare un flusso di lavoro Ready2Run. Un flusso di lavoro Ready2Run è preconfigurato con i parametri e i riferimenti agli strumenti necessari per eseguire il flusso di lavoro. L'editore del workflow fornisce dati di esempio, quindi non è necessario creare dati propri.

1. Apri la HealthOmics console.
2. Seleziona il pannello di navigazione (≡) in alto a sinistra e seleziona Ready2Run Workflows.
3. Nella pagina Flussi di lavoro Ready2Run, scegli il flusso di lavoro. ESMFold for up to 800 residues La console apre la pagina dei dettagli per quel flusso di lavoro.
4. La scheda dei dettagli fornisce informazioni sul flusso di lavoro. Per provare il flusso di lavoro, seleziona Avvia esecuzione in alto a destra della pagina.
5. Nella pagina Specificare i dettagli della corsa, inserisci un nome di esecuzione.
6. Inserisci o seleziona una posizione Amazon S3 per l'output di esecuzione.
7. Per la modalità di conservazione dei metadati Run, scegli se conservare o rimuovere i metadati runmeta.
8. Nel pannello Ruolo di servizio, scegli Crea e usa un nuovo ruolo di servizio.
9. Scegli Next (Successivo).
10. Nella pagina Aggiungi valori dei parametri, scegli Esegui flusso di lavoro con dati di test Ready2Run.
11. Scegli Next (Successivo).
12. Controlla i dati immessi, quindi scegli Avvia esecuzione.

Esempi di istruzioni per Amazon Q CLI

La CLI di Amazon Q può eseguire flussi di lavoro genomici e attività di analisi utilizzando comandi in AWS HealthOmics linguaggio naturale. I seguenti prompt di esempio consentono di creare flussi di lavoro, gestire esecuzioni e analizzare dati genomici. Per ulteriori informazioni e suggerimenti di esempio, consulta il tutorial sull'intelligenza artificiale generativa di HealthOmics Agentic su. HealthOmics GitHub

- «Crea un file di workflow WDL 1.1 su cui verrà eseguito. `main.wdl` HealthOmics Il flusso di lavoro utilizzerà un genoma di riferimento come input e coppie di file fastq. Indicizzerà il genoma di riferimento utilizzando BWA e quindi mapperà ogni coppia di file fastq sul riferimento. Infine unisci ogni BAM mappato in un singolo file BAM e restituisci questo file e il suo indice bai.»
- «Package del workflow e crealo in HealthOmics»
- «Aggiorna il file inputs.json per utilizzare file reali dal mio bucket Amazon S3" (`omics-my-bucket-with-genome-data`fornisci una posizione specifica per il bucket Amazon S3 o lascia che Amazon Q esplori)
- «Trova contenitori adatti nei miei repository Amazon ECR e aggiorna inputs.json per utilizzarli»
- «Trova o crea un ruolo IAM adatto da utilizzare durante l'esecuzione del flusso di lavoro»
- «Crea una cache di esecuzione per il mio flusso di lavoro»
- «Esegui il flusso di lavoro in HealthOmics»
- «Controlla lo stato della corsa»

> **Warning**
>
> Quando lavori con Amazon Q CLI, esamina tutti i contenuti generati e le azioni proposte prima di procedere. Fornisci feedback per migliorare la qualità della risposta e soddisfare i requisiti del tuo flusso di lavoro. Per ulteriori informazioni, consulta Considerazioni sulla sicurezza e best practice per Amazon Q.

Flussi di lavoro privati in HealthOmics

Usa i flussi di lavoro privati quando desideri creare la tua definizione di flusso di lavoro. La definizione del flusso di lavoro specifica le informazioni sul flusso di lavoro e definisce le attività del flusso di lavoro. Un'esecuzione è una singola chiamata di un flusso di lavoro e un'attività è un singolo processo all'interno dell'esecuzione.

HealthOmics supporta le definizioni dei flussi di lavoro create in Workflow Description Language (WDL), Common Workflow Language (CWL) o Nextflow.

HealthOmics i flussi di lavoro offrono le seguenti funzionalità opzionali:

- Run groups— È possibile aggiungere flussi di lavoro privati a un gruppo di esecuzione per controllare l'utilizzo dell'elaborazione. Un gruppo di esecuzione è una raccolta di esecuzioni di workflow che condividono una serie di limiti di risorse, ad esempio il numero massimo di esecuzioni simultanee e la durata massima dell'esecuzione. Questi limiti vengono impostati per controllare le risorse di elaborazione utilizzate dal gruppo di esecuzione.
- Call caching— È possibile utilizzare una cache delle chiamate per salvare e riutilizzare gli output delle attività, con conseguenti tempi di esecuzione più brevi e risparmi sui costi di calcolo.
- Sharing workflows— È possibile condividere i flussi di lavoro privati con altri Account AWS utenti della stessa regione.
- Workflow versions— È possibile creare versioni di un flusso di lavoro privato. Il controllo delle versioni del flusso di lavoro offre agli utenti la possibilità di scegliere quando iniziare a utilizzare funzionalità aggiornate. Le versioni dei flussi di lavoro sono immutabili e forniscono lo stesso livello di provenienza dei dati dei flussi di lavoro.

Per informazioni sulla configurazione delle autorizzazioni IAM per i flussi di lavoro, consulta. Autorizzazioni IAM per HealthOmics

Per esempi completi di come utilizzare i flussi di lavoro HealthOmics privati, consulta i tutorial di HealthOmics Github o il tutorial end-to-end del workshop AWS per. HealthOmics

Argomenti

- Creazione di flussi di lavoro privati in HealthOmics
- Controllo delle versioni del flusso di lavoro in HealthOmics
- Utilizzo delle HealthOmics corse

- Utilizzo dei gruppi di HealthOmics esecuzione
- Memorizzazione nella cache delle chiamate per le esecuzioni HealthOmics
- Condivisione dei flussi HealthOmics di lavoro

Creazione di flussi di lavoro privati in HealthOmics

I flussi di lavoro privati dipendono da una varietà di risorse create e configurate prima di creare il flusso di lavoro:

- Workflow definition file:Un file di definizione del flusso di lavoro scritto in WDLNextflow, oCWL. La definizione del flusso di lavoro specifica gli input e gli output per le esecuzioni che utilizzano il flusso di lavoro. Include inoltre le specifiche per le attività di esecuzione ed esecuzione del flusso di lavoro, inclusi i requisiti di calcolo e memoria. Il file di definizione del flusso di lavoro deve essere in `.zip` formato. Per ulteriori informazioni, vedere File di definizione del flusso di lavoro.
  - Puoi utilizzare Amazon Q CLI per creare e convalidare i file di definizione del flusso di lavoro in WDL, Nextflow e CWL. Per ulteriori informazioni, consulta le istruzioni di esempio per la CLI di Amazon Q e il tutorial sull'intelligenza artificiale generativa HealthOmics Agentic su. GitHub
- (Optional) Parameter template file:Un file modello di parametro scritto in. JSON Crea il file per definire i parametri di esecuzione o HealthOmics genera automaticamente il modello dei parametri. Per ulteriori informazioni, consultate File modello di parametri per i HealthOmics flussi di lavoro.
- Amazon ECR container images:Crea un repository Amazon ECR privato per il flusso di lavoro. Crea immagini di container nel repository privato o sincronizza il contenuto di un registro upstream supportato con il tuo repository privato Amazon ECR.
- (Optional) Sentieon licenses:Richiedi una Sentieon licenza per utilizzare il software in flussi di lavoro privatiSentieon.

Facoltativamente, è possibile eseguire un linter sulla definizione del flusso di lavoro prima o dopo la creazione del flusso di lavoro. L'linterargomento descrive i linter disponibili in. HealthOmics

Argomenti

- HealthOmics integrazione del flusso di lavoro con repository basati su Git
- File di definizione del flusso di lavoro in HealthOmics
- File modello di parametri per i flussi HealthOmics di lavoro
- Immagini di container per flussi di lavoro privati

- HealthOmics File README del flusso di lavoro
- Richiesta di licenze Sentieon per flussi di lavoro privati
- Stampanti per flussi di lavoro in HealthOmics
- HealthOmics operazioni del flusso di lavoro

HealthOmics integrazione del flusso di lavoro con repository basati su Git

Quando crei un flusso di lavoro (o una versione del flusso di lavoro), fornisci una definizione del flusso di lavoro per specificare informazioni sul flusso di lavoro, sulle esecuzioni e sulle attività. HealthOmics può recuperare la definizione del flusso di lavoro come archivio.zip (archiviato localmente o in un bucket Amazon S3) o da un repository basato su Git supportato.

L' HealthOmics integrazione con i repository basati su Git abilita le seguenti funzionalità:

- Creazione diretta di flussi di lavoro da istanze pubbliche, private e autogestite.
- Integrazione di file README e modelli di parametri del flusso di lavoro dai repository.
- Support per GitHub e GitLab repository Bitbucket.

Utilizzando un repository basato su Git, eviti i passaggi manuali di download dei file di definizione del flusso di lavoro e dei file modello dei parametri di input, la creazione di un archivio.zip e quindi l'archiviazione temporanea su S3. Ciò semplifica la creazione di flussi di lavoro per scenari come i seguenti esempi:

1. Vuoi iniziare rapidamente a utilizzare un flusso di lavoro open source comune, come nf-core. HealthOmicsrecupera automaticamente tutti i file di definizione del flusso di lavoro e dei modelli dei parametri di input dal repository nf-core in poi GitHub e utilizza questi file per creare il nuovo flusso di lavoro.
2. Stai utilizzando un flusso di lavoro pubblico da GitHub e saranno disponibili alcuni nuovi aggiornamenti. È possibile creare facilmente una nuova versione del HealthOmics flusso di lavoro utilizzando la definizione aggiornata del flusso di lavoro GitHub come origine. Gli utenti del flusso di lavoro possono scegliere tra il flusso di lavoro originale o la nuova versione del flusso di lavoro che hai creato.
3. Il tuo team sta creando una pipeline proprietaria che non è pubblica. Conservi il codice in un repository git privato e usi questa definizione di flusso di lavoro per i tuoi flussi di lavoro. HealthOmics Il team aggiorna frequentemente la definizione del flusso di lavoro come parte di un

ciclo di vita iterativo di sviluppo del flusso di lavoro. È possibile creare facilmente nuove versioni del flusso di lavoro in base alle esigenze dal proprio archivio privato.

Argomenti

Repository basati su Git supportati

HealthOmics supporta repository pubblici e privati per i seguenti provider basati su Git:

- GitHub
- GitLab
- Bitbucket

HealthOmics supporta repository autogestiti per i seguenti provider basati su Git:

- GitHubEnterpriseServer
- GitLabSelfManaged

HealthOmics supporta l'uso di connessioni tra account per GitHub, e Bitbucket. GitLab Configura le autorizzazioni condivise tramite AWS Resource Access Manager. Per un esempio, consulta Connessioni condivise nella guida per l'CodePipeline utente.

Configura le connessioni a repository di codice esterni

Connect i tuoi flussi di lavoro a repository basati su Git utilizzando AWS. CodeConnection HealthOmics utilizza questa connessione per accedere ai tuoi repository di codice sorgente.

> **Note**
>
> Il CodeConnections servizio AWS non è disponibile nella regione il-central-1. Per questa regione, configura il servizio us-east-1 per creare flussi di lavoro o versioni del flusso di lavoro da un repository.

Creazione di una connessione

Prima di creare connessioni, segui le istruzioni in Configurazione delle connessioni nella Guida per l'utente degli strumenti della Developer Console.

Per creare una connessione, segui le istruzioni riportate in Creare una connessione nella Developer Console Tools User Guide.

Configura l'autorizzazione per la connessione

È necessario autorizzare la connessione utilizzando il OAuth flusso del provider. Assicurati che lo stato della connessione sia valido `AVAILABLE` prima di utilizzarla.

Per esempi, consulta il post del blog Come creare un AWS HealthOmics flusso di lavoro dal contenuto in Git.

Accesso agli archivi autogestiti

Per configurare le connessioni a un repository GitLab autogestito, utilizza un token di accesso personale di amministrazione durante la creazione di un host. La successiva creazione della connessione accede a Oauth con l'account del cliente.

L'esempio seguente configura una connessione a un repository autogestito: GitLab

1. Configura l'accesso al token di accesso personale di un utente amministratore.

   Per configurare un PAT in un repository GitLab autogestito, vedi Token di accesso personali in Docs. GitLab
2. Creazione di un host

   a. Vai a >Impostazioni>Connessioni. CodePipeline
   b. Scegli la scheda Host, quindi scegli Crea host.
   c. Configura i campi seguenti:

- Inserisci il nome dell'host
- Per il tipo di provider, scegli GitLab Self Managed
- Inserisci l'URL dell'host
- Inserisci le informazioni sul VPC se l'host è definito in un VPC

d. Scegli Create Host, che crea l'host nello stato PENDING.

e. Per completare la configurazione, scegli Configura host.

f. Inserisci il Personal Access Token (PAT) di un utente amministratore, quindi scegli Continua.

3. Crea la connessione

a. Scegli Crea connessioni nella scheda Connessioni.

b. Per il tipo di provider, seleziona GitLab Autogestito.

c. In Impostazioni di connessione > Inserisci il nome della connessione, inserisci l'URL dell'host che hai creato in precedenza.

d. Se l'istanza GitLab autogestita è accessibile solo tramite un VPC, configura i dettagli del VPC.

e. Scegli Aggiorna connessione in sospeso. La finestra modale ti reindirizza alla pagina di accesso. GitLab

f. Inserisci il nome utente e la password per l'account cliente e completa il processo di autorizzazione.

g. Per la prima configurazione, scegli Authorize AWS Connector for Gitlab Self Managed.

Quote relative a repository di codice esterni

Per HealthOmics l'integrazione con gli archivi di codice esterni, è prevista una dimensione massima per un repository, ogni file di repository e ogni file README. Per informazioni dettagliate, vedi HealthOmics quote a dimensione fissa per il flusso di lavoro.

Autorizzazioni IAM richieste

Aggiungi le seguenti azioni alla tua policy IAM basata sull'identità:

```
"codeconnections:CreateConnection",
"codeconnections:GetConnection",
"codeconnections:GetHost",
```

```
    "codeconnections:ListConnections",
    "codeconnections:UseConnection"
```

File di definizione del flusso di lavoro in HealthOmics

Si utilizza una definizione di flusso di lavoro per specificare informazioni sul flusso di lavoro, sulle esecuzioni e sulle attività in corso. Le definizioni del flusso di lavoro vengono create in uno o più file utilizzando un linguaggio di definizione del flusso di lavoro. HealthOmics supporta le definizioni dei flussi di lavoro scritte in WDL, Nextflow o CWL.

HealthOmics supporta le seguenti scelte per le definizioni dei flussi di lavoro WDL:

- WDL — Fornisce un motore WDL conforme alle specifiche.
- WDL lenient: progettato per gestire i flussi di lavoro migrati da Cromwell. Supporta le direttive Cromwell dei clienti e alcune logiche non conformi. Per informazioni dettagliate, vedi Conversione implicita dei tipi in WDL lenient.

Per informazioni su ciascuna delle lingue del flusso di lavoro, consulta le sezioni dettagliate specifiche della lingua di seguito.

Nella definizione del flusso di lavoro si specificano i seguenti tipi di informazioni:

- Language version— La lingua e la versione della definizione del flusso di lavoro.
- Compute and memory— I requisiti di calcolo e memoria per le attività del flusso di lavoro.
- Inputs— Ubicazione degli input per le attività del flusso di lavoro. Per ulteriori informazioni, consulta HealthOmics eseguire ingressi.
- Outputs— Posizione in cui salvare gli output generati dalle attività.
- Task resources— Requisiti di calcolo e memoria per ogni attività.
- Accelerators— altre risorse richieste dalle attività, come gli acceleratori.

Argomenti

- Le risorse delle attività in una definizione di HealthOmics flusso di lavoro
- Acceleratori di attività in una definizione di workflow HealthOmics
- Specifiche della definizione del flusso di lavoro WDL
- Specifiche della definizione del flusso di lavoro Nextflow
- Specifiche della definizione del flusso di lavoro CWL
- Esempi di definizioni del flusso di lavoro

HealthOmics requisiti di definizione del flusso di lavoro

I file HealthOmics di definizione del flusso di lavoro devono soddisfare i seguenti requisiti:

- Le attività devono definire input/output parametri, repository di contenitori Amazon ECR e specifiche di runtime come l'allocazione della memoria o della CPU.
- Verifica che i tuoi ruoli IAM dispongano delle autorizzazioni richieste.
  - Il tuo flusso di lavoro ha accesso ai dati di input provenienti da AWS risorse, come Amazon S3.
  - Il tuo flusso di lavoro ha accesso a servizi di repository esterni quando necessario.
- Dichiara i file di output nella definizione del flusso di lavoro. Per copiare i file di esecuzione intermedi nella posizione di output, dichiarateli come output del flusso di lavoro.
- Le posizioni di input e output devono trovarsi nella stessa regione del flusso di lavoro.
- HealthOmics gli input del flusso di lavoro di archiviazione devono essere in `ACTIVE` stato. HealthOmics non importerà input con uno `ARCHIVED` stato, causando il fallimento del flusso di lavoro. Per informazioni sugli input degli oggetti Amazon S3, consulta. HealthOmics eseguire ingressi
- La main posizione del flusso di lavoro è facoltativa se l'archivio ZIP contiene una singola definizione di flusso di lavoro o un file denominato «principale».
  - Percorso di esempio: `workflow-definition/main-file.wdl`
- Prima di creare un flusso di lavoro da Amazon S3 o dall'unità locale, crea un archivio zip dei file di definizione del flusso di lavoro e di eventuali dipendenze, come i flussi di lavoro secondari.
- Ti consigliamo di dichiarare i contenitori Amazon ECR nel flusso di lavoro come parametri di input per la convalida delle autorizzazioni Amazon ECR.

Considerazioni aggiuntive su Nextflow:

- /bin

Le definizioni del flusso di lavoro Nextflow possono includere una cartella /bin con script eseguibili. Questo percorso ha accesso alle attività in sola lettura ed eseguibile. Le attività che si basano su questi script devono utilizzare un contenitore creato con gli interpreti di script appropriati. La migliore pratica consiste nel chiamare direttamente l'interprete. Ad esempio:

```
process my_bin_task {
   ...
   script:
      """
      python3 my_python_script.py
      """
}
```

- includeConfig

  Le definizioni dei flussi di lavoro basate su NextFlow possono includere file nextflow.config che aiutano ad astrarre le definizioni dei parametri o i profili delle risorse di processo. Per supportare lo sviluppo e l'esecuzione di pipeline Nextflow su più ambienti, utilizza una configurazione HealthOmics specifica da aggiungere alla configurazione globale utilizzando la direttiva IncludeConfig. Per mantenere la portabilità, configura il flusso di lavoro in modo che includa il file solo durante l'esecuzione utilizzando il codice seguente: HealthOmics

  ```
  // at the end of the nextflow.config file
  if ("$AWS_WORKFLOW_RUN") {
      includeConfig 'conf/omics.config'
  }
  ```

- Reports

  HealthOmics non supporta i report dag, trace ed esecuzione generati dal motore. È possibile generare alternative ai report di traccia ed esecuzione utilizzando una combinazione di GetRun chiamate API. GetRunTask

Considerazioni aggiuntive sulla CWL:

- Container image uri interpolation

HealthOmics consente alla proprietà DockerPull di essere un'espressione DockerRequirement javascript in linea. Ad esempio:

```
requirements:
  DockerRequirement:
    dockerPull: "$(inputs.container_image)"
```

Ciò consente di specificare l'immagine del contenitore URIs come parametri di input per il flusso di lavoro.

- Javascript expressions

  Le espressioni Javascript devono essere `strict mode` conformi.

- Operation process

  HealthOmics non supporta i processi operativi CWL.

Supporto della versione per i linguaggi HealthOmics di definizione del flusso di lavoro

HealthOmics supporta file di definizione del flusso di lavoro scritti in Nextflow, WDL o CWL. Le seguenti sezioni forniscono informazioni sul supporto delle HealthOmics versioni per queste lingue.

Argomenti

- Supporto della versione WDL
- Supporto per la versione CWL
- Supporto per la versione Nextflow

Supporto della versione WDL

HealthOmics supporta le versioni 1.0, 1.1 e la versione di sviluppo della specifica WDL.

Ogni documento WDL deve includere una dichiarazione di versione per specificare a quale versione (principale e secondaria) della specifica a cui aderisce. Per ulteriori informazioni sulle versioni, vedere WDL versioning

Le versioni 1.0 e 1.1 della specifica WDL non supportano questo tipo. `Directory` Per utilizzare il `Directory` tipo per input o output, impostate la versione su development nella prima riga del file:

```
version development  # first line of .wdl file
     ... remainder of the file ...
```

Supporto per la versione CWL

HealthOmics supporta le versioni 1.0, 1.1 e 1.2 del linguaggio CWL.

È possibile specificare la versione della lingua nel file di definizione del flusso di lavoro CWL. Per ulteriori informazioni su CWL, consultate la guida per l'utente di CWL

Supporto per la versione Nextflow

HealthOmics supporta tre versioni stabili di Nextflow. Nextflow rilascia in genere una versione stabile ogni sei mesi. HealthOmics non supporta le versioni mensili «edge».

HealthOmics supporta le funzionalità rilasciate in ogni versione, ma non le funzionalità di anteprima.

Versioni supportate

HealthOmics supporta le seguenti versioni di Nextflow:

- Nextflow v22.04.01 DSL 1 e DSL 2
- Nextflow v23.10.0 DSL 2 (impostazione predefinita)
- Nextflow versione 24.10.8 DSL 2

Per migrare il flusso di lavoro all'ultima versione supportata (v24.10.8), segui la guida all'aggiornamento di Nextflow.

Ci sono alcune modifiche importanti durante la migrazione da Nextflow v23 a v24, come descritto nelle seguenti sezioni della guida alla migrazione di Nextflow:

- Ultime modifiche nella versione 24.04
- Ultime modifiche nel 24.10

Rileva ed elabora le versioni di Nextflow

HealthOmics rileva la versione DSL e la versione di Nextflow specificate. Determina automaticamente la migliore versione di Nextflow da eseguire in base a questi input.

Versione DSL

HealthOmics rileva la versione DSL richiesta nel file di definizione del flusso di lavoro. Ad esempio, puoi specificare:. `nextflow.enable.dsl=2`

HealthOmics supporta DSL 2 per impostazione predefinita. Fornisce la retrocompatibilità con DSL 1, se specificato nel file di definizione del flusso di lavoro.

- Se si specifica DSL 2, HealthOmics esegue Nextflow v23.10.0, a meno che non si specifichi Nextflow v22.04.0 o v24.10.8.
- Se si specifica DSL 1, esegue Nextflow v22.04 (l'unica versione supportata che esegue DSL 1). HealthOmics DSL1
- Se non specifichi una versione DSL o se non HealthOmics riesci ad analizzare le informazioni DSL per qualsiasi motivo (ad esempio errori di sintassi nel file di definizione del flusso di lavoro), il valore predefinito è DSL 2 ed esegue Nextflow v23.10.0. HealthOmics
- Per aggiornare il flusso di lavoro da DSL 1 a DSL 2 per sfruttare le versioni e le funzionalità software più recenti di Nextflow, consulta Migrazione da DSL 1.

Versioni Nextflow

HealthOmics rileva la versione di Nextflow richiesta nel file di configurazione di Nextflow (nextflow.config), se fornisci questo file. Ti consigliamo di aggiungere la `nextflowVersion` clausola alla fine del file per evitare sostituzioni impreviste delle configurazioni incluse. Per ulteriori informazioni, consulta la configurazione di Nextflow.

Puoi specificare una versione di Nextflow o un intervallo di versioni utilizzando la seguente sintassi:

```
// exact match
manifest.nextflowVersion = '1.2.3'

// 1.2 or later (excluding 2 and later)
manifest.nextflowVersion = '1.2+'

// 1.2 or later
manifest.nextflowVersion = '>=1.2'

// any version in the range 1.2 to 1.5
manifest.nextflowVersion = '>=1.2, <=1.5'
```

```
// use the "!" prefix to stop execution if the current version
// doesn't match the required version.
manifest.nextflowVersion = '!>=1.2'
```

HealthOmics elabora le informazioni sulla versione di Nextflow come segue:

- Se si utilizza = per specificare una versione esatta che HealthOmics supporta, HealthOmics utilizza quella versione.
- Se si utilizza ! per specificare una versione esatta o un intervallo di versioni non supportate, HealthOmics genera un'eccezione e fallisce l'esecuzione. Prendi in considerazione l'utilizzo di questa opzione se desideri essere rigoroso con le richieste di versione e fallire rapidamente se la richiesta include versioni non supportate.
- Se specifichi un intervallo di versioni, HealthOmics utilizza l'ultima versione supportata in quell'intervallo, a meno che l'intervallo non includa la v24.10.8. In questo caso, HealthOmics dà la preferenza a una versione precedente. Ad esempio, se l'intervallo copre sia la v23.10.0 che la v24.10.8, sceglie v23.10.0. HealthOmics
- Se non esiste una versione richiesta o se le versioni richieste non sono valide o non possono essere analizzate per qualsiasi motivo:
  - Se hai specificato DSL 1, HealthOmics esegue Nextflow v22.04.
  - Altrimenti, esegue Nextflow v23.10.0. HealthOmics

Puoi recuperare le seguenti informazioni sulla versione di Nextflow utilizzata per ogni esecuzione: HealthOmics

- I log di esecuzione contengono informazioni sulla versione effettiva di Nextflow utilizzata per l'esecuzione. HealthOmics
- HealthOmics aggiunge avvisi nei log di esecuzione se non esiste una corrispondenza diretta con la versione richiesta o se è necessario utilizzare una versione diversa da quella specificata.
- La risposta all'operazione GetRun API include un campo (`engineVersion`) con la versione effettiva di Nextflow HealthOmics utilizzata per l'esecuzione. Per esempio:

```
"engineVersion":"22.04.0"
```

Requisiti di calcolo e memoria per le attività HealthOmics

HealthOmics esegue le attività private del flusso di lavoro in un'istanza omics. HealthOmics offre una varietà di tipi di istanze per soddisfare diversi tipi di attività. Ogni tipo di istanza ha una configurazione fissa di memoria e vCPU (e una configurazione GPU fissa per i tipi di istanze di elaborazione accelerata). Il costo dell'utilizzo di un'istanza omics varia a seconda del tipo di istanza. Per i dettagli, consulta la pagina HealthOmics dei prezzi.

Per le attività in un flusso di lavoro, si specifica la memoria richiesta e v CPUs nel file di definizione del flusso di lavoro. Quando viene eseguita un'operazione di workflow, HealthOmics alloca l'istanza omics più piccola che contiene la memoria richiesta e v. CPUs Ad esempio, se un'attività richiede 64 GiB di memoria e 8 vCPUs, HealthOmics seleziona. `omics.r.2xlarge`

Ti consigliamo di esaminare i tipi di istanza e di impostare la v CPUs e la dimensione di memoria richieste in modo che corrispondano all'istanza che meglio soddisfa le tue esigenze. Il contenitore delle attività utilizza il numero di v CPUs e la dimensione della memoria specificati nel file di definizione del flusso di lavoro, anche se il tipo di istanza dispone di v CPUs e memoria aggiuntivi.

L'elenco seguente contiene informazioni aggiuntive su vCPU e allocazione della memoria:

- Le allocazioni di risorse dei container sono limiti rigidi. Se un'attività esaurisce la memoria o tenta di utilizzare v aggiuntiviCPUs , l'attività genera un registro degli errori e viene chiusa.
- Se non specifichi alcun requisito di calcolo o memoria, HealthOmics seleziona omics.c.large e imposta per impostazione predefinita una configurazione con 1 vCPU e 1 GiB di memoria.
- La configurazione minima che è possibile richiedere è 1 vCPU e 1 GiB di memoria.
- Se si specifica vCPUs, memory o GPUs che supera i tipi di istanza supportati, viene generato un messaggio di errore e il HealthOmics flusso di lavoro non viene convalidato
- Se specificate unità frazionarie, HealthOmics arrotondate al numero intero più vicino.
- HealthOmics riserva una piccola quantità di memoria (5%) per gli agenti di gestione e registrazione, pertanto l'allocazione completa della memoria potrebbe non essere sempre disponibile per l'applicazione incaricata del task.
- HealthOmics corrisponde ai tipi di istanza per soddisfare i requisiti di calcolo e memoria specificati e può utilizzare una combinazione di generazioni hardware. Per questo motivo, possono verificarsi alcune lievi variazioni nei tempi di esecuzione delle attività per la stessa attività.

Questi argomenti forniscono dettagli sui tipi di istanze HealthOmics supportati.

Argomenti

> **Note**
>
> Per le istanze standard, di calcolo e ottimizzate per la memoria, aumenta la dimensione della larghezza di banda dell'istanza se l'istanza richiede un throughput più elevato. Le istanze Amazon EC2 con meno di 16 vCPU (dimensioni 4xl e inferiori) possono subire un aumento del throughput. Per ulteriori informazioni sulla velocità effettiva delle istanze Amazon EC2, consulta Larghezza di banda disponibile delle istanze di Amazon EC2.

Tipi di istanze standard

Per i tipi di istanze standard, le configurazioni mirano a un equilibrio tra potenza di calcolo e memoria.

HealthOmics supporta le istanze 32xlarge e 48xlarge in queste regioni: Stati Uniti occidentali (Oregon) e Stati Uniti orientali (Virginia settentrionale).

Istanza	Numero di v CPUs	Memoria
omics.m.large	2	8 GiB
omics.m.xlarge	4	16 GiB
omics.m.2xlarge	8	32 GiB
omics.m.4xlarge	16	64 GiB
omics.m.8xlarge	32	128 GiB
omics.m.12xlarge	48	192 GiB
omics.m.16xlarge	64	256 GiB

Istanza	Numero di v CPUs	Memoria
omics.m.24xlarge	96	384 GiB
omics.m.32xlarge	128	512 GiB
omics.m.48xlarge	192	768 GiB

Istanze ottimizzate per il calcolo

Per i tipi di istanza ottimizzati per l'elaborazione, le configurazioni offrono maggiore potenza di calcolo e meno memoria.

HealthOmics supporta le istanze 32xlarge e 48xlarge in queste regioni: Stati Uniti occidentali (Oregon) e Stati Uniti orientali (Virginia settentrionale).

Istanza	Numero di v CPUs	Memoria
omics.c.large	2	4 GiB
omics.c.xlarge	4	8 GiB
omics.c.2xlarge	8	16 GiB
omics.c.4xlarge	16	32 GiB
omics.c.8xlarge	32	64 GiB
omics.c.12xlarge	48	96 GiB
omics.c. 16 x large	64	128 GiB
omics.c.24xlarge	96	192 GiB
omics.c.32xlarge	128	256 GiB
omics.c.48xlarge	192	384 GiB

Istanze ottimizzate per la memoria

Per i tipi di istanza ottimizzati per la memoria, le configurazioni hanno meno potenza di calcolo e più memoria.

HealthOmics supporta le istanze 32xlarge e 48xlarge in queste regioni: Stati Uniti occidentali (Oregon) e Stati Uniti orientali (Virginia settentrionale).

Istanza	Numero di v CPUs	Memoria
omic s.r.large	2	16 GiB
omic s.r.xlarge	4	32 GiB
omic s.r.2xlarge	8	64 GiB
omic s.r.4 x grande	16	128 GiB
omic s.r.8xlarge	32	256 GiB
omic s.r.12 x grande	48	384 GiB
omics.r. 16 x grande	64	512 GiB
omic s.r.24xlarge	96	768 GiB
omic s.r.32xlarge	128	1024 GiB
Fomics.r. 48 x grande	192	1536 GiB

Istanze di elaborazione accelerata

Facoltativamente, è possibile specificare risorse GPU per ogni attività in un flusso di lavoro, in modo da HealthOmics allocare un'istanza di elaborazione accelerata per l'attività. Per informazioni su come specificare le informazioni sulla GPU nel file di definizione del flusso di lavoro, vedere. Acceleratori di attività in una definizione di workflow HealthOmics

Se si specifica una GPU che supporta più tipi di istanza, HealthOmics seleziona il tipo di istanza in base alla disponibilità. Se entrambi i tipi di istanza sono disponibili, HealthOmics dà la preferenza all'istanza a basso costo.

Le istanze G4 non sono supportate nella regione di Israele (Tel Aviv). Le istanze G5 non sono supportate nella regione Asia Pacifico (Singapore).

Argomenti

Tipi di istanze G6 e G6e

HealthOmics supporta le seguenti configurazioni di istanza di calcolo accelerato G6. Tutte le istanze omics.g6 utilizzano Nvidia L4 o Nvidia L4 A10G. GPUs

HealthOmics supporta le istanze G6 e G6e in queste regioni: Stati Uniti occidentali (Oregon) e Stati Uniti orientali (Virginia settentrionale).

Istanza	Numero di v CPUs	Memoria	Numero di GPUs	Memoria GPU	
omics.g6.xlarge	4	16 GiB	1	24 GiB	
omics.g6.2xlarge	8	32 GiB	1	24 GiB	
omics.g6.4xlarge	16	64 GiB	1	24 GiB	
omics.g6.8xlarge	32	128 GiB	1	24 GiB	
omics.g6.12xlarge	48	192 GiB	4	96 GiB	
omics.g6.16xlarge	64	256 GiB	1	24 GiB	
omics.g6.24xlarge	96	384 GiB	4	96 GiB	

Tutte le istanze omics.g6e utilizzano Nvidia L40s. GPUs

Istanza	Numero di v CPUs	Memoria	Numero di GPUs	Memoria GPU	
omics.g6e .xlarge	4	32 GiB	1	48 GiB	
omics.g6e .2xlarge	8	64 GiB	1	48 GiB	
omics.g6e .4xlarge	16	128 GiB	1	48 GiB	
omics.g6e .8xlarge	32	256 GiB	1	48 GiB	
omics.g6e .12xlarge	48	384 GiB	4	192 GiB	
omics.g6e .16xlarge	64	512 GiB	1	48 GiB	
omics.g6e .24xlarge	96	768 GiB	4	192 GiB	

Istanze G4 e G5

HealthOmics supporta le seguenti configurazioni di istanze di calcolo accelerato G4 e G5.

Tutte le istanze omics.g5 utilizzano Nvidia L4 A10G, Nvidia Tesla A10G o Nvidia Tesla T4 A10G. GPUs

Istanza	Numero di v CPUs	Memoria	Numero di GPUs	Memoria GPU	
omics.g5. xlarge	4	16 GiB	1	24 GiB	

Istanza	Numero di v CPUs	Memoria	Numero di GPUs	Memoria GPU	
omics.g5.2xlarge	8	32 GiB	1	24 GiB	
omics.g5.4xlarge	16	64 GiB	1	24 GiB	
omics.g5.8xlarge	32	128 GiB	1	24 GiB	
omics.g5.12xlarge	48	192 GiB	4	96 GiB	
omics.g5.16xlarge	64	256 GiB	1	24 GiB	
omics.g5.24xlarge	96	384 GiB	4	96 GiB	

Tutte le istanze omics.g4dn utilizzano Nvidia Tesla T4 o Nvidia Tesla T4 A10G. GPUs

Istanza	Numero di v CPUs	Memoria	Numero di GPUs	Memoria GPU	
omics.g4dn.xlarge	4	16 GiB	1	16 GiB	
omics.g4dn.2xlarge	8	32 GiB	1	16 GiB	
omics.g4dn.4xlarge	16	64 GiB	1	16 GiB	
omics.g4dn.8xlarge	32	128 GiB	1	16 GiB	

Istanza	Numero di v CPUs	Memoria	Numero di GPUs	Memoria GPU	
omics.g4dn.12xlarge	48	192 GiB	4	64 GiB	
omics.g4dn.16xlarge	64	256 GiB	1	24 GiB	

Risultati delle attività in una definizione di HealthOmics flusso di lavoro

Gli output delle attività vengono specificati nella definizione del flusso di lavoro. Per impostazione predefinita, HealthOmics elimina tutti i file di attività intermedi al termine del flusso di lavoro. Per esportare un file intermedio, lo si definisce come output.

Se si utilizza la memorizzazione nella cache delle chiamate, HealthOmics salva gli output delle attività nella cache, inclusi tutti i file intermedi definiti come output.

I seguenti argomenti includono esempi di definizione delle attività per ciascuno dei linguaggi di definizione del flusso di lavoro.

Argomenti

Output delle attività per WDL

Per le definizioni dei flussi di lavoro scritte in WDL, definite i risultati nella sezione Workflow di primo livello. outputs

HealthOmics

Argomenti

- Emissione dell'operazione su una serie di file

Output delle attività per STDOUT

Questo esempio crea un'attività denominata `SayHello` che riproduce il contenuto STDOUT nel file di output dell'operazione. La stdout funzione WDL acquisisce il contenuto STDOUT (in questo esempio, la stringa di input Hello World!) nel file. stdout_file

Poiché HealthOmics crea registri per tutto il contenuto STDOUT, l'output viene visualizzato anche in CloudWatch Logs, insieme ad altre informazioni di registrazione STDERR relative all'attività.

```
version 1.0
 workflow HelloWorld {
    input {
        String message = "Hello, World!"
        String ubuntu_container = "123456789012.dkr.ecr.us-east-1.amazonaws.com/
dockerhub/library/ubuntu:20.04"
    }

    call SayHello {
        input:
            message = message,
            container = ubuntu_container
    }

    output {
        File stdout_file = SayHello.stdout_file
    }
}

task SayHello {
    input {
        String message
        String container
    }

    command <<<
        echo "~{message}"
        echo "Current date: $(date)"
        echo "This message was printed to STDOUT"
    >>>
```

```
    runtime {
        docker: container
        cpu: 1
        memory: "2 GB"
    }

    output {
        File stdout_file = stdout()
    }
}
```

Output dell'attività per STDERR

Questo esempio crea un'attività denominata `SayHello` che riproduce il contenuto STDERR del file di output dell'operazione. La stderr funzione WDL acquisisce il contenuto STDERR (in questo esempio, la stringa di input Hello World!) nel file. stderr_file

Poiché HealthOmics crea registri per tutto il contenuto STDERR, l'output verrà visualizzato in CloudWatch Logs, insieme ad altre informazioni di registrazione STDERR relative all'attività.

```
version 1.0
 workflow HelloWorld {
    input {
        String message = "Hello, World!"
        String ubuntu_container = "123456789012.dkr.ecr.us-east-1.amazonaws.com/
dockerhub/library/ubuntu:20.04"
    }

    call SayHello {
        input:
            message = message,
            container = ubuntu_container
    }

    output {
        File stderr_file = SayHello.stderr_file
    }
}

task SayHello {
    input {
        String message
        String container
```

```
    }

    command <<<
        echo "~{message}" >&2
        echo "Current date: $(date)" >&2
        echo "This message was printed to STDERR" >&2
    >>>

    runtime {
        docker: container
        cpu: 1
        memory: "2 GB"
    }

    output {
        File stderr_file = stderr()
    }
}
```

Emissione dell'operazione in un file

In questo esempio, l' SayHello attività crea due file (message.txt e info.txt) e li dichiara esplicitamente come output denominato (message_file e info_file).

```
version 1.0
workflow HelloWorld {
    input {
        String message = "Hello, World!"
        String ubuntu_container = "123456789012.dkr.ecr.us-east-1.amazonaws.com/
dockerhub/library/ubuntu:20.04"
    }

    call SayHello {
        input:
            message = message,
            container = ubuntu_container
    }

    output {
        File message_file = SayHello.message_file
        File info_file = SayHello.info_file
    }
}
```

```
task SayHello {
    input {
        String message
        String container
    }

    command <<<
        # Create message file
        echo "~{message}" > message.txt

        # Create info file with date and additional information
        echo "Current date: $(date)" > info.txt
        echo "This message was saved to a file" >> info.txt
    >>>

    runtime {
        docker: container
        cpu: 1
        memory: "2 GB"
    }

    output {
        File message_file = "message.txt"
        File info_file = "info.txt"
    }
}
```

Emissione dell'operazione su una serie di file

In questo esempio, l'`GenerateGreetings`attività genera una serie di file come output dell'operazione. L'operazione genera dinamicamente un file di saluto per ogni membro dell'array di input. `names` Poiché i nomi dei file non sono noti fino all'esecuzione, la definizione di output utilizza la funzione WDL glob () per generare tutti i file che corrispondono al modello. `*_greeting.txt`

```
version 1.0
 workflow HelloArray {
    input {
        Array[String] names = ["World", "Friend", "Developer"]
        String ubuntu_container = "123456789012.dkr.ecr.us-east-1.amazonaws.com/
dockerhub/library/ubuntu:20.04"
    }
```

```
    call GenerateGreetings {
        input:
            names = names,
            container = ubuntu_container
    }

    output {
        Array[File] greeting_files = GenerateGreetings.greeting_files
    }
}

task GenerateGreetings {
    input {
        Array[String] names
        String container
    }

    command  <<<
        # Create a greeting file for each name
        for name in ~{sep=" " names}; do
            echo "Hello, $name!" > ${name}_greeting.txt
        done
    >>>

    runtime {
        docker: container
        cpu: 1
        memory: "2 GB"
    }

    output {
        Array[File] greeting_files = glob("*_greeting.txt")
    }
 }
```

Output delle attività per Nextflow

Per le definizioni dei flussi di lavoro scritte in Nextflow, definisci una direttiva PublishDir per esportare il contenuto delle attività nel tuo bucket Amazon S3 di output. **`/mnt/workflow/pubdir`**Imposta il valore PublishDir su.

HealthOmics Per esportare file in Amazon S3, i file devono trovarsi in questa directory.

Se un'operazione produce un gruppo di file di output da utilizzare come input per un'attività successiva, consigliamo di raggruppare questi file in una directory e di emettere la directory come output dell'attività. L'enumerazione di ogni singolo file può causare un collo di bottiglia di I/O nel file system sottostante. Per esempio:

```
process my_task {
      ...
      // recommended
      output "output-folder/", emit: output

      // not recommended
      // output "output-folder/**", emit: output
      ...
  }
```

Output delle attività per CWL

Per le definizioni dei flussi di lavoro scritte in CWL, è possibile specificare gli output delle attività utilizzando le attività. `CommandLineTool` Le sezioni seguenti mostrano esempi di `CommandLineTool` attività che definiscono diversi tipi di output.

Argomenti

- Output delle attività per STDOUT
- Output dell'attività per STDERR
- Emissione dell'operazione in un file
- Esecuzione dell'operazione su una serie di file

Output delle attività per STDOUT

Questo esempio crea un'`CommandLineTool`attività che riproduce il contenuto STDOUT in un file di output di testo denominato. output.txt Ad esempio, se fornite il seguente input, l'output dell'attività risultante è Hello World! nel output.txt file.

```
{
    "message": "Hello World!"
}
```

La `outputs` direttiva specifica che il nome dell'output è example_out e il suo tipo è`stdout`. Affinché un'attività a valle consumi l'output di questa attività, si riferirebbe all'output come. `example_out`

Poiché HealthOmics crea registri per tutto il contenuto STDERR e STDOUT, l'output viene visualizzato anche in CloudWatch Logs, insieme ad altre informazioni di registrazione STDERR relative all'attività.

```
cwlVersion: v1.2
class: CommandLineTool
baseCommand: echo
stdout: output.txt
inputs:
  message:
    type: string
    inputBinding:
      position: 1
outputs:
  example_out:
    type: stdout

requirements:
    DockerRequirement:
        dockerPull: 123456789012.dkr.ecr.us-east-1.amazonaws.com/dockerhub/library/
ubuntu:20.04
    ResourceRequirement:
        ramMin: 2048
        coresMin: 1
```

Output dell'attività per STDERR

Questo esempio crea un'`CommandLineTool`attività che riproduce il contenuto STDERR in un file di output di testo denominato. stderr.txt L'attività modifica il codice in `baseCommand` modo che venga `echo` scritto su STDERR (anziché su STDOUT).

La `outputs` direttiva specifica che il nome dell'output è e il tipo èstderr_out. `stderr`

Poiché HealthOmics crea registri per tutto il contenuto STDERR e STDOUT, l'output verrà visualizzato in CloudWatch Logs, insieme ad altre informazioni di registrazione STDERR relative all'attività.

```
cwlVersion: v1.2
class: CommandLineTool
baseCommand: [bash, -c]
stderr: stderr.txt
inputs:
```

```
  message:
    type: string
    inputBinding:
      position: 1
      shellQuote: true
      valueFrom: "echo $(self) >&2"
outputs:
  stderr_out:
    type: stderr

requirements:
    DockerRequirement:
        dockerPull: 123456789012.dkr.ecr.us-east-1.amazonaws.com/dockerhub/library/
ubuntu:20.04
    ResourceRequirement:
        ramMin: 2048
        coresMin: 1
```

Emissione dell'operazione in un file

Questo esempio crea un'`CommandLineTool`attività che crea un archivio tar compresso dai file di input. Fornite il nome dell'archivio come parametro di input (archive_name).

La outputs direttiva specifica che il tipo di `archive_file` output è `File` e utilizza un riferimento al parametro di input per collegarsi `archive_name` al file di output.

```
cwlVersion: v1.2
class: CommandLineTool
baseCommand: [tar, cfz]
inputs:
  archive_name:
    type: string
    inputBinding:
      position: 1
  input_files:
    type: File[]
    inputBinding:
      position: 2

outputs:
  archive_file:
    type: File
    outputBinding:
```

```
      glob: "$(inputs.archive_name)"

requirements:
    DockerRequirement:
        dockerPull: 123456789012.dkr.ecr.us-east-1.amazonaws.com/dockerhub/library/
ubuntu:20.04
    ResourceRequirement:
        ramMin: 2048
        coresMin: 1
```

Esecuzione dell'operazione su una serie di file

In questo esempio, l'`CommandLineTool`attività crea una matrice di file utilizzando il `touch` comando. Il comando utilizza le stringhe del parametro `files-to-create` di input per denominare i file. Il comando genera una serie di file. L'array include tutti i file nella directory di lavoro che corrispondono al `glob` modello. Questo esempio utilizza un pattern di caratteri jolly («*») che corrisponde a tutti i file.

```
cwlVersion: v1.2
class: CommandLineTool
baseCommand: touch
inputs:
  files-to-create:
    type:
      type: array
      items: string
    inputBinding:
      position: 1
outputs:
  output-files:
    type:
      type: array
      items: File
    outputBinding:
      glob: "*"

requirements:
    DockerRequirement:
        dockerPull: 123456789012.dkr.ecr.us-east-1.amazonaws.com/dockerhub/library/
ubuntu:20.04
    ResourceRequirement:
        ramMin: 2048
```

```
        coresMin: 1
```

Le risorse delle attività in una definizione di HealthOmics flusso di lavoro

Nella definizione del flusso di lavoro, definisci quanto segue per ogni attività:

- L'immagine del contenitore per l'attività. Per ulteriori informazioni, consulta Immagini di container per flussi di lavoro privati.
- Il numero CPUs e la memoria necessari per l'operazione. Per ulteriori informazioni, consulta Requisiti di calcolo e memoria per le attività HealthOmics.

HealthOmics ignora le specifiche di archiviazione relative alle singole attività. HealthOmics fornisce un'archiviazione di esecuzione a cui possono accedere tutte le attività in esecuzione. Per ulteriori informazioni, consulta Esegui tipi di storage nei flussi HealthOmics di lavoro.

WDL

```
task my_task {
   runtime {
      container: "<aws-account-id>.dkr.ecr.<aws-region>.amazonaws.com/<image-name>"
      cpu: 2
      memory: "4 GB"
   }
   ...
}
```

Per un flusso di lavoro WDL, HealthOmics tenta fino a due nuovi tentativi per un'operazione che non riesce a causa di errori di servizio (la richiesta API restituisce un codice di stato HTTP 5XX). Per ulteriori informazioni sui nuovi tentativi di attività, vedere. Ritentativi di attività

È possibile disattivare il comportamento dei nuovi tentativi specificando la seguente configurazione per l'operazione nel file di definizione WDL:

```
runtime {
   preemptible: 0
}
```

NextFlow

```
process my_task {
```

```
    container "<aws-account-id>.dkr.ecr.<aws-region>.amazonaws.com/<image-name>"
    cpus 2
    memory "4 GiB"
    ...
}
```

CWL

```
cwlVersion: v1.2
class: CommandLineTool
requirements:
    DockerRequirement:
        dockerPull: "<aws-account-id>.dkr.ecr.<aws-region>.amazonaws.com/<image-
name>"
    ResourceRequirement:
        coresMax: 2
        ramMax: 4000 # specified in mebibytes
```

Acceleratori di attività in una definizione di workflow HealthOmics

Nella definizione del flusso di lavoro, puoi facoltativamente specificare le specifiche dell'acceleratore GPU per un'attività. HealthOmics supporta i seguenti valori accelerator-spec, oltre ai tipi di istanza supportati:

Specifiche dell'acceleratore	Tipi di istanze Healthomics				
nvidia-tesla-t4	G4				
nvidia-tesla-t4 a 10 g	G4 e G5				
nvidia-tesla-a10 g	G5				

Specifiche dell'acceleratore	Tipi di istanze Healthomics
nvidia-l4-a10g	G5 e G6
nvidia-l4	G6
nvidia-l40s	G6e

Se si specifica un tipo di acceleratore che supporta più tipi di istanza, HealthOmics seleziona il tipo di istanza in base alla capacità disponibile. Se entrambi i tipi di istanza sono disponibili, HealthOmics dà la preferenza all'istanza a basso costo.

Per informazioni dettagliate sui tipi di istanza, consultaIstanze di elaborazione accelerata.

Nell'esempio seguente, la definizione del flusso di lavoro specifica `nvidia-l4` come acceleratore:

WDL

```
task my_task {
 runtime {
    ...
    acceleratorCount: 1
    acceleratorType: "nvidia-l4"
 }
 ...
}
```

NextFlow

```
process my_task {
 ...
 accelerator 1, type: "nvidia-l4"
 ...
}
```

CWL

```
cwlVersion: v1.2
class: CommandLineTool
requirements:
  ...
  cwltool:CUDARequirement:
      cudaDeviceCountMin: 1
      cudaComputeCapability: "nvidia-l4"
      cudaVersionMin: "1.0"
```

Specifiche della definizione del flusso di lavoro WDL

I seguenti argomenti forniscono dettagli sui tipi e le direttive disponibili per le definizioni dei flussi di lavoro WDL in. HealthOmics

Argomenti

Conversione implicita dei tipi in WDL lenient

HealthOmics supporta la conversione implicita dei tipi nel file input.json e nella definizione del flusso di lavoro. Per utilizzare il tipo di casting implicito, specificate il motore di workflow come WDL clemente quando create il flusso di lavoro. WDL lenient è progettato per gestire i flussi di lavoro migrati da Cromwell. Supporta le direttive Cromwell dei clienti e alcune logiche non conformi.

WDL lenient supporta la conversione dei tipi per i seguenti elementi nell'elenco delle eccezioni limitate di WDL:

- Float to Int, dove la coercizione non comporta alcuna perdita di precisione (ad esempio 1,0 mappato a 1).
- String to Int/Float, dove la coercizione non comporta alcuna perdita di precisione.
- Mappa [W, X] su Array [Pair [Y, Z]], nel caso in cui W sia coercibile con Y e X sia coercibile con Z.
- Array [Pair [W, X] to Map [Y, Z], nel caso in cui W sia coercibile con Y e X sia coercibile con Z (ad esempio 1,0 mappato a 1).

Per utilizzare il type casting implicito, specificate il motore di workflow come WDL_LENIENT quando create il workflow o la versione del workflow.

Nella console, il parametro del motore di workflow è denominato Language. Nell'API, il parametro del motore di workflow è denominato engine. Per ulteriori informazioni, consulta Creare un flusso di lavoro privato o Creare una versione del flusso di lavoro.

Definizione dello spazio dei nomi in input.json

HealthOmics supporta variabili completamente qualificate in input.json. Ad esempio, se dichiarate due variabili di input denominate number1 e number2 nel flusso di lavoro: SumWorkflow

```
workflow SumWorkflow {
  input {
    Int number1
    Int number2
  }
}
```

Puoi usarle come variabili complete in input.json:

```
{
    "SumWorkflow.number1": 15,
    "SumWorkflow.number2": 27
}
```

Tipi primitivi in WDL

La tabella seguente mostra come gli input in WDL vengono mappati ai tipi primitivi corrispondenti. HealthOmics fornisce un supporto limitato per la coercizione dei tipi, quindi consigliamo di impostare tipi espliciti.

tipi primitivi

Tipo WDL	Tipo JSON	Esempio WDL	Esempio di chiave e valore JSON	Note
`Boolean`	`boolean`	`Boolean b`	`"b": true`	Il valore deve essere minuscolo e senza virgolette.
`Int`	`integer`	`Int i`	`"i": 7`	Deve essere senza virgolette.
`Float`	`number`	`Float f`	`"f": 42.2`	Deve essere non quotato.
`String`	`string`	`String s`	`"s": "characters"`	Le stringhe JSON che sono un URI devono essere mappate su un file WDL per essere importate.
`File`	`string`	`File f`	`"f": "s3://amzn-s3-demo-bucket1/path/to/file"`	Amazon S3 e HealthOmics lo storage URIs vengono importati purché il ruolo IAM fornito per il flusso di lavoro abbia accesso in lettura a questi oggetti. Non sono supportati i altri schemi

Tipo WDL	Tipo JSON	Esempio WDL	Esempio di chiave e valore JSON	Note
				URI (come `file://https://`, `eftp://`). L'URI deve specificare un oggetto. Non può essere una directory, il che significa che non può terminare con un`/`.

Tipo WDL	Tipo JSON	Esempio WDL	Esempio di chiave e valore JSON	Note
`Directory`	`string`	`Directory d`	`"d": "s3://bucket/path/"`	Il `Directory` tipo non è incluso in WDL 1.0 o 1.1, quindi sarà necessario aggiungere lo `version development` all'intestazione del file WDL. L'URI deve essere un URI Amazon S3 e deve avere un prefisso che termina con un '/'. Tutti i contenuti della directory verranno copiati in modo ricorsivo nel flusso di lavoro come singolo download. `Directory` Deve contenere solo file relativi al flusso di lavoro.

Tipi complessi in WDL

La tabella seguente mostra come gli input in WDL vengono mappati ai tipi JSON complessi corrispondenti. I tipi complessi in WDL sono strutture di dati composte da tipi primitivi. Le strutture di dati, come gli elenchi, verranno convertite in matrici.

Tipi complessi

Tipo WDL	Tipo JSON	Esempio WDL	Esempio di chiave e valore JSON	Note
`Array`	`array`	`Array[Int] nums`	`“nums": [1, 2, 3]`	I membri dell'array devono seguire il formato del tipo di array WDL.
`Pair`	`object`	`Pair[String, Int] str_to_i`	`“str_to_i": {"left": "0", "right": 1}`	Ogni valore della coppia deve utilizzare il formato JSON del tipo WDL corrispondente.
`Map`	`object`	`Map[Int, String] int_to_string`	`"int_to_string": { 2: "hello", 1: "goodbye" }`	Ogni voce nella mappa deve utilizzare il formato JSON del tipo WDL corrispondente.
`Struct`	`object`	`struct SampleBamAndIndex { String sample_name File bam File bam_index`	`"b_and_i": { "sample_name": "NA12878", "bam": "s3://amzn-s3-demo`	I nomi dei membri della struttura devono corrispondere esattamente ai nomi delle chiavi degli oggetti JSON.

Tipo WDL	Tipo JSON	Esempio WDL	Esempio di chiave e valore JSON	Note
		`} SampleBam AndIndex b_and_i`	`-bucket1/ NA12878.b am", "bam_inde x": "s3:// amzn- s3-demo- bucket1/ NA12878.b am.bai" }`	Ogni valore deve utilizzare il formato JSON del tipo WDL corrispondente.
`Object`	N/D	N/D	N/D	Il `Object` tipo WDL è obsoleto e deve essere sostituito da `Struct` in tutti i casi.

Direttive in WDL

HealthOmics supporta le seguenti direttive in tutte le versioni WDL che supportano. HealthOmics

Configura le risorse della GPU

HealthOmics supporta gli attributi di runtime acceleratorType e acceleratorCount con tutte le istanze GPU supportate. HealthOmics supporta anche gli alias denominati gpuType andgpuCount, che hanno le stesse funzionalità delle controparti con acceleratore. Se la definizione WDL contiene entrambe le direttive, HealthOmics utilizza i valori dell'acceleratore.

L'esempio seguente mostra come utilizzare queste direttive:

```
runtime {
    gpuCount: 2
    gpuType: "nvidia-tesla-t4"
}
```

Configurare la ripetizione delle attività per gli errori di servizio

HealthOmics supporta fino a due tentativi per un'operazione non riuscita a causa di errori di servizio (codici di stato HTTP 5XX). È possibile configurare il numero massimo di tentativi (1 o 2) e disattivare i tentativi per errori di servizio. Per impostazione predefinita, HealthOmics tenta un massimo di due nuovi tentativi.

L'esempio seguente imposta `preemptible` la disattivazione dei nuovi tentativi per errori di servizio:

```
{
  preemptible: 0
}
```

Per ulteriori informazioni sui nuovi tentativi di attività in HealthOmics, vedere. Ritentativi di attività

Configurare un nuovo tentativo di operazione in caso di esaurimento della memoria

HealthOmics supporta nuovi tentativi per un'operazione che non è riuscita perché la memoria è esaurita (codice di uscita del contenitore 137, codice di stato HTTP 4XX). HealthOmics raddoppia la quantità di memoria per ogni nuovo tentativo.

Per impostazione predefinita, HealthOmics non riprova per questo tipo di errore. Utilizzate la `maxRetries` direttiva per specificare il numero massimo di tentativi.

L'esempio seguente è `maxRetries` impostato su 3, in modo che HealthOmics tenti un massimo di quattro tentativi di completare l'operazione (il tentativo iniziale più tre tentativi):

```
runtime {
    maxRetries: 3
}
```

> **Note**
>
> Per riprovare l'operazione se la memoria non è sufficiente, è necessario GNU findutils 4.2.3+. Il contenitore di immagini predefinito include questo pacchetto. HealthOmics Se specificate un'immagine personalizzata nella definizione WDL, assicuratevi che l'immagine includa GNU findutils 4.2.3+.

Configura i codici di restituzione

L'attributo ReturnCodes fornisce un meccanismo per specificare un codice di ritorno, o un set di codici di ritorno, che indica l'esecuzione corretta di un'attività. Il motore WDL rispetta i codici restituiti specificati nella sezione runtime della definizione WDL e imposta lo stato delle attività di conseguenza.

```
runtime {
    returnCodes: 1
}
```

HealthOmics supporta anche un alias denominato continueOnReturnCode, che ha le stesse funzionalità di ReturnCodes. Se si specificano entrambi gli attributi, HealthOmics utilizza il valore ReturnCodes.

Metadati delle attività in WDL

HealthOmics supporta le seguenti opzioni di metadati per le attività WDL.

Disattiva la memorizzazione nella cache a livello di attività con l'attributo volatile

L'attributo volatile consente di disabilitare la memorizzazione nella cache delle chiamate per attività specifiche nel flusso di lavoro WDL. Quando un'attività è contrassegnata come volatile, verrà sempre eseguita e non utilizzerà mai i risultati memorizzati nella cache, anche quando la memorizzazione nella cache è abilitata per l'esecuzione.

Aggiungi l'attributo volatile alla meta-sezione della definizione dell'attività:

```
task my_volatile_task {
    meta {
        volatile: true
    }

    input {
        String input_file
    }

    command {
        echo "Processing ${input_file}" > output.txt
    }
```

```
    output {
        File result = "output.txt"
    }
}
```

Esempio di definizione del flusso di lavoro WDL

Gli esempi seguenti mostrano le definizioni di workflow private per la conversione da `CRAM` a `BAM` in WDL. Il `BAM` flusso di lavoro `CRAM` to definisce due attività e utilizza gli strumenti del `genomes-in-the-cloud` contenitore, illustrato nell'esempio e disponibile pubblicamente.

L'esempio seguente mostra come includere il contenitore Amazon ECR come parametro. Ciò consente di HealthOmics verificare le autorizzazioni di accesso al contenitore prima che inizi l'esecuzione.

```
{
   ...
   "gotc_docker":"<account_id>.dkr.ecr.<region>.amazonaws.com/genomes-in-the-
cloud:2.4.7-1603303710"
}
```

L'esempio seguente mostra come specificare quali file utilizzare nell'esecuzione, quando i file si trovano in un bucket Amazon S3.

```
{
    "input_cram": "s3://amzn-s3-demo-bucket1/inputs/NA12878.cram",
    "ref_dict": "s3://amzn-s3-demo-bucket1/inputs/Homo_sapiens_assembly38.dict",
    "ref_fasta": "s3://amzn-s3-demo-bucket1/inputs/Homo_sapiens_assembly38.fasta",
    "ref_fasta_index": "s3://amzn-s3-demo-bucket1/inputs/
Homo_sapiens_assembly38.fasta.fai",
    "sample_name": "NA12878"
}
```

Se desideri specificare file da un archivio di sequenze, indicalo come mostrato nell'esempio seguente, utilizzando l'URI per l'archivio delle sequenze.

```
{
    "input_cram": "omics://429915189008.storage.us-west-2.amazonaws.com/111122223333/
readSet/4500843795/source1",
    "ref_dict": "s3://amzn-s3-demo-bucket1/inputs/Homo_sapiens_assembly38.dict",
    "ref_fasta": "s3://amzn-s3-demo-bucket1/inputs/Homo_sapiens_assembly38.fasta",
```

```
    "ref_fasta_index": "s3://amzn-s3-demo-bucket1/inputs/
Homo_sapiens_assembly38.fasta.fai",
    "sample_name": "NA12878"
}
```

È quindi possibile definire il flusso di lavoro in WDL come illustrato nell'esempio seguente.

```
 version 1.0
workflow CramToBamFlow {
    input {
        File ref_fasta
        File ref_fasta_index
        File ref_dict
        File input_cram
        String sample_name
        String gotc_docker = "<account>.dkr.ecr.us-west-2.amazonaws.com/genomes-in-the-
cloud:latest"
    }
    #Converts CRAM to SAM to BAM and makes BAI.
    call CramToBamTask{
         input:
            ref_fasta = ref_fasta,
            ref_fasta_index = ref_fasta_index,
            ref_dict = ref_dict,
            input_cram = input_cram,
            sample_name = sample_name,
            docker_image = gotc_docker,
     }
     #Validates Bam.
     call ValidateSamFile{
        input:
           input_bam = CramToBamTask.outputBam,
           docker_image = gotc_docker,
     }
     #Outputs Bam, Bai, and validation report to the FireCloud data model.
     output {
         File outputBam = CramToBamTask.outputBam
         File outputBai = CramToBamTask.outputBai
         File validation_report = ValidateSamFile.report
      }
}
#Task definitions.
task CramToBamTask {
```

```
    input {
       # Command parameters
       File ref_fasta
       File ref_fasta_index
       File ref_dict
       File input_cram
       String sample_name
       # Runtime parameters
       String docker_image
    }
   #Calls samtools view to do the conversion.
   command {
       set -eo pipefail

       samtools view -h -T ~{ref_fasta} ~{input_cram} |
       samtools view -b -o ~{sample_name}.bam -
       samtools index -b ~{sample_name}.bam
       mv ~{sample_name}.bam.bai ~{sample_name}.bai
    }

    #Runtime attributes:
    runtime {
        docker: docker_image
    }

    #Outputs a BAM and BAI with the same sample name
     output {
         File outputBam = "~{sample_name}.bam"
         File outputBai = "~{sample_name}.bai"
    }
}

#Validates BAM output to ensure it wasn't corrupted during the file conversion.
task ValidateSamFile {
   input {
      File input_bam
      Int machine_mem_size = 4
      String docker_image
   }
   String output_name = basename(input_bam, ".bam") + ".validation_report"
   Int command_mem_size = machine_mem_size - 1
   command {
       java -Xmx~{command_mem_size}G -jar /usr/gitc/picard.jar \
       ValidateSamFile \
```

```
        INPUT=~{input_bam} \
        OUTPUT=~{output_name} \
        MODE=SUMMARY \
        IS_BISULFITE_SEQUENCED=false
    }
    runtime {
    docker: docker_image
    }
   #A text file is generated that lists errors or warnings that apply.
    output {
        File report = "~{output_name}"
    }
}
```

Specifiche della definizione del flusso di lavoro Nextflow

HealthOmics supporta Nextflow e. DSL1 DSL2 Per informazioni dettagliate, vedi Supporto per la versione Nextflow.

Nextflow si DSL2 basa sul linguaggio di programmazione Groovy, quindi i parametri sono dinamici e la coercizione dei tipi è possibile utilizzando le stesse regole di Groovy. I parametri e i valori forniti dall'input JSON sono disponibili nella mappa parameters () del flusso di lavoro. `params`

Argomenti

- Usa i plugin nf-schema e nf-validation
- Specificare l'archiviazione URIs
- Direttive Nextflow
- Esporta il contenuto delle attività

Usa i plugin nf-schema e nf-validation

> **Note**
>
> Riepilogo del supporto per i plugin: HealthOmics
>
> - v22.04 — nessun supporto per i plugin
> - v23.10 — supporta e `nf-schema` `nf-validation`
> - v24.10 — supporta `nf-schema`

HealthOmics fornisce il seguente supporto per i plugin Nextflow:

- Per Nextflow v23.10, preinstalla il plugin nf-validation @1 .1.1 HealthOmics .
- Per Nextflow v23.10 e versioni successive, preinstalla il plugin nf-schema @2 .3.0. HealthOmics
- Non è possibile recuperare plug-in aggiuntivi durante l'esecuzione di un flusso di lavoro. HealthOmics ignora qualsiasi altra versione del plugin specificata nel file. `nextflow.config`
- Per Nextflow v24 e versioni successive, `nf-schema` è la nuova versione del plugin obsoleto. `nf-validation` Per ulteriori informazioni, consulta nf-schema nel repository Nextflow. GitHub

Specificare l'archiviazione URIs

Quando un Amazon S3 o un HealthOmics URI viene utilizzato per costruire un file o un oggetto di percorso Nextflow, rende l'oggetto corrispondente disponibile per il flusso di lavoro, purché sia concesso l'accesso in lettura. L'uso di prefissi o directory è consentito per Amazon S3. URIs Per alcuni esempi, consulta Formati dei parametri di input di Amazon S3.

HealthOmics supporta parzialmente l'uso di modelli glob in Amazon URIs S3 HealthOmics o Storage. URIs Usa i pattern Glob nella definizione del flusso di lavoro per la creazione dei `path` nostri canali. `file` Per il comportamento previsto e i casi esatti, vediGestione Nextflow del pattern Glob negli input di Amazon S3.

Direttive Nextflow

Le direttive Nextflow vengono configurate nel file di configurazione di Nextflow o nella definizione del flusso di lavoro. L'elenco seguente mostra l'ordine di precedenza HealthOmics utilizzato per applicare le impostazioni di configurazione, dalla priorità più bassa a quella più alta:

1. Configurazione globale nel file di configurazione.
2. Sezione delle attività della definizione del flusso di lavoro.
3. Selettori specifici dell'attività nel file di configurazione.

Argomenti

- Strategia di ripetizione delle attività utilizzando errorStrategy
- Tentativi di riprovare l'attività utilizzando maxRetries
- Disattiva la ripetizione dell'attività utilizzando omicsRetryOn5xx
- timeDurata dell'attività utilizzando la direttiva

Strategia di ripetizione delle attività utilizzando **errorStrategy**

Usa la `errorStrategy` direttiva per definire la strategia per gli errori delle attività. Per impostazione predefinita, quando un'attività ritorna con un'indicazione di errore (uno stato di uscita diverso da zero), l'attività si interrompe e HealthOmics termina l'intera esecuzione. Se è impostata su `errorStrategyretry`, HealthOmics tenta un nuovo tentativo dell'operazione non riuscita. Per aumentare il numero di tentativi, vedere. Tentativi di riprovare l'attività utilizzando `maxRetries`

```
process {
    label 'my_label'
    errorStrategy 'retry'

    script:
    """
    your-command-here
    """
}
```

Per informazioni su come HealthOmics gestisce i nuovi tentativi di operazione durante un'esecuzione, vedere. Ritentativi di attività

Tentativi di riprovare l'attività utilizzando **maxRetries**

Per impostazione predefinita, HealthOmics non tenta di ripetere un'operazione non riuscita o tenta un solo tentativo se configurato. `errorStrategy` Per aumentare il numero massimo di tentativi, imposta `errorStrategy` `retry` e configura il numero massimo di tentativi utilizzando la direttiva. `maxRetries`

L'esempio seguente imposta il numero massimo di tentativi su 3 nella configurazione globale.

```
process {
    errorStrategy = 'retry'
    maxRetries = 3
}
```

L'esempio seguente mostra come impostare `maxRetries` la definizione del flusso di lavoro nella sezione delle attività.

```
process myTask {
    label 'my_label'
    errorStrategy 'retry'
```

```
    maxRetries 3

    script:
    """
    your-command-here
    """
}
```

L'esempio seguente mostra come specificare la configurazione specifica dell'attività nel file di configurazione di Nextflow, in base ai selettori di nome o etichetta.

```
process {
    withLabel: 'my_label' {
        errorStrategy = 'retry'
        maxRetries = 3
    }

    withName: 'myTask' {
        errorStrategy = 'retry'
        maxRetries = 3
    }
}
```

Disattiva la ripetizione dell'attività utilizzando **`omicsRetryOn5xx`**

Per Nextflow v23 e v24, HealthOmics supporta nuovi tentativi di attività se l'operazione non è riuscita a causa di errori di servizio (5XX codici di stato HTTP). Per impostazione predefinita, HealthOmics tenta fino a due nuovi tentativi di un'operazione non riuscita.

È possibile `omicsRetryOn5xx` configurare la disattivazione del nuovo tentativo di operazione per errori di servizio. Per ulteriori informazioni su come riprovare l'attività HealthOmics, vedere. Ritentativi di attività

L'esempio seguente configura `omicsRetryOn5xx` la configurazione globale per disattivare il nuovo tentativo di operazione.

```
process {
    omicsRetryOn5xx = false
}
```

L'esempio seguente mostra come eseguire la configurazione `omicsRetryOn5xx` nella sezione delle attività della definizione del flusso di lavoro.

```
process myTask {
    label 'my_label'
    omicsRetryOn5xx = false

    script:
    """
    your-command-here
    """
}
```

L'esempio seguente mostra `omicsRetryOn5xx` come impostare una configurazione specifica per l'attività nel file di configurazione Nextflow, in base ai selettori di nome o etichetta.

```
process {
    withLabel: 'my_label' {
        omicsRetryOn5xx = false
    }

    withName: 'myTask' {
        omicsRetryOn5xx = false
    }
}
```

`time`Durata dell'attività utilizzando la direttiva

HealthOmics fornisce una quota regolabile (vediHealthOmics quote di servizio) per specificare la durata massima di un'esecuzione. Per i flussi di lavoro Nextflow v23 e v24, puoi anche specificare la durata massima delle attività utilizzando la direttiva Nextflow. `time`

Durante lo sviluppo di nuovi flussi di lavoro, l'impostazione della durata massima delle attività consente di catturare attività inutili e attività di lunga durata.

Per ulteriori informazioni sulla direttiva time di Nextflow, consulta la direttiva time nel riferimento di Nextflow.

HealthOmics fornisce il seguente supporto per la direttiva time Nextflow:

1. HealthOmics supporta la granularità di 1 minuto per la direttiva time. È possibile specificare un valore compreso tra 60 secondi e il valore massimo della durata dell'esecuzione.
2. Se si immette un valore inferiore a 60, lo HealthOmics arrotonda a 60 secondi. Per valori superiori a 60, HealthOmics arrotonda per difetto al minuto più vicino.

3. Se il flusso di lavoro supporta nuovi tentativi per un'operazione, HealthOmics riprova l'operazione in caso di timeout.
4. Se un'attività scade (o scade l'ultimo tentativo), HealthOmics annulla l'attività. Questa operazione può avere una durata da uno a due minuti.
5. In caso di timeout dell'attività, HealthOmics imposta l'esecuzione e lo stato dell'attività su Non riuscita e annulla le altre attività in esecuzione (per le attività con stato Avvio, In sospeso o In esecuzione). HealthOmics esporta gli output delle attività completate prima del timeout nella posizione di output S3 designata.
6. Il tempo trascorso da un'attività in sospeso non viene conteggiato ai fini della durata dell'attività.
7. Se l'esecuzione fa parte di un gruppo di esecuzione e il gruppo di esecuzione scade prima del timer dell'attività, l'esecuzione e l'attività passano allo stato non riuscito.

Specificate la durata del timeout utilizzando una o più delle seguenti unità:`ms`,, `s` `mh`, o. `d`

L'esempio seguente mostra come specificare la configurazione globale nel file di configurazione Nextflow. Imposta un timeout globale di 1 ora e 30 minuti.

```
process {
    time = '1h30m'
}
```

L'esempio seguente mostra come specificare una direttiva temporale nella sezione delle attività della definizione del flusso di lavoro. Questo esempio imposta un timeout di 3 giorni, 5 ore e 4 minuti. Questo valore ha la precedenza sul valore globale nel file di configurazione, ma non ha la precedenza su una direttiva temporale specifica per l'attività nel file di configurazione. `my_label`

```
process myTask {
    label 'my_label'
    time '3d5h4m'

    script:
    """
    your-command-here
    """
}
```

L'esempio seguente mostra come specificare le direttive temporali specifiche dell'attività nel file di configurazione Nextflow, in base ai selettori di nome o etichetta. Questo esempio imposta un valore

di timeout globale dell'attività di 30 minuti. Imposta un valore di 2 ore per l'attività `myTask` e imposta un valore di 3 ore per le attività con etichettamy_`label`. Per le attività che corrispondono al selettore, questi valori hanno la precedenza sul valore globale e sul valore nella definizione del flusso di lavoro.

```
process {
    time = '30m'

    withLabel: 'my_label' {
        time = '3h'
    }

    withName: 'myTask' {
        time = '2h'
    }
}
```

Esporta il contenuto delle attività

Per i flussi di lavoro scritti in Nextflow, definisci una direttiva PublishDir per esportare il contenuto delle attività nel tuo bucket Amazon S3 di output. Come mostrato nell'esempio seguente, imposta il valore PublishDir su. `/mnt/workflow/pubdir` Per esportare file in Amazon S3, i file devono trovarsi in questa directory.

```
 nextflow.enable.dsl=2

  workflow {
    CramToBamTask(params.ref_fasta, params.ref_fasta_index, params.ref_dict,
params.input_cram, params.sample_name)
    ValidateSamFile(CramToBamTask.out.outputBam)
  }

  process CramToBamTask {
    container "<account>.dkr.ecr.us-west-2.amazonaws.com/genomes-in-the-cloud"

    publishDir "/mnt/workflow/pubdir"

    input:
        path ref_fasta
        path ref_fasta_index
        path ref_dict
        path input_cram
        val sample_name
```

```
  output:
      path "${sample_name}.bam", emit: outputBam
      path "${sample_name}.bai", emit: outputBai

  script:
  """
      set -eo pipefail

      samtools view -h -T $ref_fasta $input_cram |
      samtools view -b -o ${sample_name}.bam -
      samtools index -b ${sample_name}.bam
      mv ${sample_name}.bam.bai ${sample_name}.bai
  """
}

process ValidateSamFile {
  container "<account>.dkr.ecr.us-west-2.amazonaws.com/genomes-in-the-cloud"

  publishDir "/mnt/workflow/pubdir"

  input:
      file input_bam

  output:
      path "validation_report"

  script:
  """
      java -Xmx3G -jar /usr/gitc/picard.jar \
      ValidateSamFile \
      INPUT=${input_bam} \
      OUTPUT=validation_report \
      MODE=SUMMARY \
      IS_BISULFITE_SEQUENCED=false
  """
}
```

Specifiche della definizione del flusso di lavoro CWL

I flussi di lavoro scritti in Common Workflow Language, o CWL, offrono funzionalità simili ai flussi di lavoro scritti in WDL e Nextflow. Puoi utilizzare Amazon S3 o HealthOmics lo storage URIs come parametri di input.

Se definisci l'input in un SecondaryFile in un flusso di lavoro secondario, aggiungi la stessa definizione nel flusso di lavoro principale.

HealthOmics i flussi di lavoro non supportano i processi operativi. Per ulteriori informazioni sui processi operativi nei flussi di lavoro CWL, consultate la documentazione CWL.

La migliore pratica consiste nel definire un flusso di lavoro CWL separato per ogni contenitore utilizzato. Ti consigliamo di non codificare la voce DockerPull con un URI Amazon ECR fisso.

Argomenti

Converti i flussi di lavoro CWL da utilizzare HealthOmics

Per convertire una definizione di workflow CWL esistente da utilizzare HealthOmics, apportate le seguenti modifiche:

- Sostituisci tutti i container Docker URIs con Amazon URIs ECR.
- Assicurati che tutti i file del flusso di lavoro siano dichiarati come input nel flusso di lavoro principale e che tutte le variabili siano definite in modo esplicito.
- Assicurati che tutto il JavaScript codice sia conforme alla modalità rigorosa.

Disattiva la ripetizione dell'attività utilizzando **`omicsRetryOn5xx`**

HealthOmics supporta nuovi tentativi di attività se l'operazione non è riuscita a causa di errori di servizio (codici di stato HTTP 5XX). Per impostazione predefinita, HealthOmics tenta fino a due nuovi tentativi di un'operazione non riuscita. Per ulteriori informazioni su come riprovare l'operazione HealthOmics, vedere. Ritentativi di attività

Per disattivare la ripetizione dell'attività a causa di errori di servizio, configura la `omicsRetryOn5xx` direttiva nella definizione del flusso di lavoro. È possibile definire questa direttiva in base a requisiti o suggerimenti. Consigliamo di aggiungere la direttiva come suggerimento per la portabilità.

```
requirements:
  ResourceRequirement:
    omicsRetryOn5xx: false

hints:
  ResourceRequirement:
    omicsRetryOn5xx: false
```

I requisiti hanno la precedenza sui suggerimenti. Se l'implementazione di un'attività fornisce un fabbisogno di risorse nei suggerimenti che è fornito anche dai requisiti in un flusso di lavoro che lo include, i requisiti di inclusione hanno la precedenza.

Se lo stesso requisito di attività viene visualizzato a livelli diversi del flusso di lavoro, HealthOmics utilizza la voce più specifica di `requirements` (o`hints`, se non sono presenti voci). `requirements` L'elenco seguente mostra l'ordine di precedenza HealthOmics utilizzato per applicare le impostazioni di configurazione, dalla priorità più bassa a quella più alta:

- Livello di flusso di lavoro
- Livello di gradino
- Sezione delle attività della definizione del flusso di lavoro

L'esempio seguente mostra come configurare la o`micsRetryOn5xx` direttiva a diversi livelli del flusso di lavoro. In questo esempio, il requisito a livello di flusso di lavoro ha la precedenza sui suggerimenti a livello di flusso di lavoro. Le configurazioni dei requisiti a livello di attività e fase hanno la precedenza sulle configurazioni dei suggerimenti.

```
class: Workflow
# Workflow-level requirement and hint
requirements:
  ResourceRequirement:
    omicsRetryOn5xx: false

hints:
  ResourceRequirement:
    omicsRetryOn5xx: false  # The value in requirements overrides this value

steps:
  task_step:
    # Step-level requirement
    requirements:
```

```
      ResourceRequirement:
        omicsRetryOn5xx: false
    # Step-level hint
    hints:
      ResourceRequirement:
        omicsRetryOn5xx: false
    run:
      class: CommandLineTool
      # Task-level requirement
      requirements:
        ResourceRequirement:
          omicsRetryOn5xx: false
      # Task-level hint
      hints:
        ResourceRequirement:
          omicsRetryOn5xx: false
```

Ripeti una fase del flusso di lavoro

HealthOmics supporta la ripetizione ciclica di una fase del flusso di lavoro. È possibile utilizzare i loop per eseguire ripetutamente le fasi del flusso di lavoro fino a quando non viene soddisfatta una condizione specificata. Ciò è utile per i processi iterativi in cui è necessario ripetere un'operazione più volte o fino al raggiungimento di un determinato risultato.

Nota: la funzionalità Loop richiede la versione CWL 1.2 o successiva. I flussi di lavoro che utilizzano versioni CWL precedenti alla 1.2 non supportano le operazioni di loop.

Per utilizzare i loop nel flusso di lavoro CWL, definite un requisito di Loop. L'esempio seguente mostra la configurazione dei requisiti del loop:

```
requirements:
  - class: "http://commonwl.org/cwltool#Loop"
    loopWhen: $(inputs.counter < inputs.max)
    loop:
      counter:
        loopSource: result
        valueFrom: $(self)
    outputMethod: last
```

Il `loopWhen` campo controlla quando il ciclo termina. In questo esempio, il ciclo continua finché il contatore è inferiore al valore massimo. Il `loop` campo definisce come i parametri di input vengono

aggiornati tra le iterazioni. `loopSource`specifica quale output dell'iterazione precedente viene immesso nell'iterazione successiva. Il `outputMethod` campo impostato su `last` restituisce solo l'output dell'iterazione finale.

Riprova le attività con maggiore memoria

HealthOmics supporta la ripetizione automatica degli errori delle out-of-memory attività. Quando un'attività termina con il codice 137 (out-of-memory), HealthOmics crea una nuova attività con una maggiore allocazione di memoria in base al moltiplicatore specificato.

> **Note**
>
> HealthOmics riprova gli out-of-memory errori fino a 3 volte o finché l'allocazione della memoria non raggiunge i 1536 GiB, a seconda del limite raggiunto per primo.

L'esempio seguente mostra come configurare retry: out-of-memory

```
hints:
  ResourceRequirement:
    ramMin: 4096
  http://arvados.org/cwl#OutOfMemoryRetry:
    memoryRetryMultiplier: 2.5
```

Quando un'operazione fallisce a causa di out-of-memory, HealthOmics calcola l'allocazione della memoria tra tentativi utilizzando la formula:. `previous_run_memory × memoryRetryMultiplier` Nell'esempio precedente, se l'operazione con 4096 MB di memoria ha esito negativo, il nuovo tentativo utilizza 4096 × 2,5 = 10.240 MB di memoria.

Il `memoryRetryMultiplier` parametro controlla la quantità di memoria aggiuntiva da allocare per i tentativi di riprovare:

- Valore predefinito: se non si specifica un valore, il valore predefinito è 2 (raddoppia la memoria)
- Intervallo valido: deve essere un numero positivo maggiore di. 1 I valori non validi generano un errore di convalida 4XX
- Valore minimo effettivo: i valori compresi tra 1 e `1.5` vengono aumentati automaticamente per `1.5` garantire un aumento significativo della memoria e prevenire tentativi eccessivi di nuovi tentativi

Esempi

Di seguito è riportato un esempio di workflow scritto in CWL.

```
cwlVersion: v1.2
class: Workflow

inputs:
in_file:
type: File
secondaryFiles: [.fai]

out_filename: string
docker_image: string


outputs:
copied_file:
type: File
outputSource: copy_step/copied_file

steps:
copy_step:
in:
  in_file: in_file
  out_filename: out_filename
  docker_image: docker_image
out: [copied_file]
run: copy.cwl
```

Il file seguente definisce l'`copy.cwl`attività.

```
cwlVersion: v1.2
class: CommandLineTool
baseCommand: cp

inputs:
in_file:
type: File
secondaryFiles: [.fai]
```

```
inputBinding:
  position: 1

out_filename:
type: string
inputBinding:
  position: 2
docker_image:
type: string

outputs:
copied_file:
type: File
outputBinding:
    glob: $(inputs.out_filename)

requirements:
InlineJavascriptRequirement: {}
DockerRequirement:
dockerPull: "$(inputs.docker_image)"
```

Di seguito è riportato un esempio di flusso di lavoro scritto in CWL con un requisito GPU.

```
cwlVersion: v1.2
class: CommandLineTool
baseCommand: ["/bin/bash", "docm_haplotypeCaller.sh"]
$namespaces:
cwltool: http://commonwl.org/cwltool#
requirements:
cwltool:CUDARequirement:
cudaDeviceCountMin: 1
cudaComputeCapability: "nvidia-tesla-t4"
cudaVersionMin: "1.0"
InlineJavascriptRequirement: {}
InitialWorkDirRequirement:
listing:
- entryname: 'docm_haplotypeCaller.sh'
  entry: |
          nvidia-smi --query-gpu=gpu_name,gpu_bus_id,vbios_version --format=csv

inputs: []
outputs: []
```

Esempi di definizioni del flusso di lavoro

L'esempio seguente mostra la stessa definizione di flusso di lavoro in WDL, Nextflow e CWL.

WDL

```
version 1.1

task my_task {
   runtime { ... }
   inputs {
       File input_file
       String name
       Int threshold
   }

   command <<<
   my_tool --name ~{name} --threshold ~{threshold} ~{input_file}
   >>>

   output {
       File results = "results.txt"
   }
}

workflow my_workflow {
   inputs {
       File input_file
       String name
       Int threshold = 50
   }

   call my_task {
       input:
          input_file = input_file,
          name = name,
          threshold = threshold
   }
   outputs {
       File results = my_task.results
   }
}
```

Nextflow

```
nextflow.enable.dsl = 2

params.input_file = null
params.name = null
params.threshold = 50

process my_task {
   // <directives>

   input:
     path input_file
     val name
     val threshold

   output:
     path 'results.txt', emit: results

   script:
     """
     my_tool --name ${name} --threshold ${threshold} ${input_file}
     """


}

workflow MY_WORKFLOW {
   my_task(
       params.input_file,
       params.name,
       params.threshold
   )
}

workflow {
   MY_WORKFLOW()
}
```

CWL

```
cwlVersion: v1.2
class: Workflow

requirements:
    InlineJavascriptRequirement: {}

inputs:
   input_file: File
   name: string
   threshold: int

outputs:
    result:
        type: ...
        outputSource: ...

steps:
    my_task:
        run:
            class: CommandLineTool
            baseCommand: my_tool
            requirements:
                ...
            inputs:
                name:
                    type: string
                    inputBinding:
                        prefix: "--name"
                threshold:
                    type: int
                    inputBinding:
                        prefix: "--threshold"
                input_file:
                    type: File
                    inputBinding: {}
            outputs:
                results:
                    type: File
                    outputBinding:
                        glob: results.txt
```

File modello di parametri per i flussi HealthOmics di lavoro

I modelli di parametri definiscono i parametri di input per un flusso di lavoro. È possibile definire i parametri di input per rendere il flusso di lavoro più flessibile e versatile. Ad esempio, puoi definire un parametro per la posizione in Amazon S3 dei file del genoma di riferimento. I modelli di parametri possono essere forniti tramite un servizio di repository basato su Git o l'unità locale. Gli utenti possono quindi eseguire il flusso di lavoro utilizzando vari set di dati.

È possibile creare il modello di parametri per il flusso di lavoro HealthOmics oppure generare il modello di parametri automaticamente.

Il modello di parametro è un file JSON. Nel file, ogni parametro di input è un oggetto denominato che deve corrispondere al nome dell'input del flusso di lavoro. Quando si avvia un'esecuzione, se non si forniscono valori per tutti i parametri richiesti, l'esecuzione ha esito negativo.

L'oggetto del parametro di input include i seguenti attributi:

- description— Questo attributo obbligatorio è una stringa che la console visualizza nella pagina Start run. Questa descrizione viene conservata anche come metadati di esecuzione.
- optional— Questo attributo opzionale indica se il parametro di input è facoltativo. Se non si specifica il optional campo, il parametro di input è obbligatorio.

Il seguente modello di parametri di esempio mostra come specificare i parametri di input.

```
{
  "myRequiredParameter1": {
     "description": "this parameter is required",
  },
  "myRequiredParameter2": {
     "description": "this parameter is also required",
     "optional": false
  },
  "myOptionalParameter": {
     "description": "this parameter is optional",
     "optional": true
  }
}
```

Generazione di modelli di parametri

HealthOmics genera il modello di parametri analizzando la definizione del flusso di lavoro per rilevare i parametri di input. Se fornite un file modello di parametri per un flusso di lavoro, i parametri del file sostituiscono i parametri rilevati nella definizione del flusso di lavoro.

Esistono lievi differenze tra la logica di analisi dei motori CWL, WDL e Nextflow, come descritto nelle sezioni seguenti.

Argomenti

Rilevamento dei parametri per CWL

Nel motore di workflow CWL, la logica di analisi si basa sui seguenti presupposti:

- Tutti i tipi nullable supportati sono contrassegnati come parametri di input opzionali.
- Tutti i tipi supportati non nulli sono contrassegnati come parametri di input obbligatori.
- Tutti i parametri con valori predefiniti sono contrassegnati come parametri di input opzionali.
- Le descrizioni vengono estratte dalla `label` sezione della definizione del `main` flusso di lavoro. Se non `label` è specificato, la descrizione sarà vuota (una stringa vuota).

Le tabelle seguenti mostrano esempi di interpolazione CWL. Per ogni esempio, il nome del parametro è. x Se il parametro è obbligatorio, è necessario fornire un valore per il parametro. Se il parametro è facoltativo, non è necessario fornire un valore.

Questa tabella mostra esempi di interpolazione CWL per tipi primitivi.

Input	Esempio di input/output	Richiesto
`x:` `  type: int`	1 o 2 o...	Sì
`x:` `  type: int`	Il valore predefinito è 2. L'input valido è 1 o 2 o...	No

Input	Esempio di input/output	Richiesto
`default: 2`		
`x:` `  type: int?`	L'input valido è Nessuno o 1 o 2 o...	No

`x:`
`  type: int?`
`  default: 2`
	Il valore predefinito è 2. L'input valido è Nessuno o 1 o 2 o...	No

La tabella seguente mostra esempi di interpolazione CWL per tipi complessi. Un tipo complesso è una raccolta di tipi primitivi.

Input	Esempio di input/output	Richiesto

`x:`
`  type: array`
`  items: int`
	[] o [1,2,3]	Sì

`x:`
`  type: array?`
`  items: int`
	Nessuno oppure [] o [1,2,3]	No

`x:`
`  type: array`
`  items: int?`
	[] o [Nessuno, 3, Nessuno]	Sì

`x:`
`  type: array?`
`  items: int?`
	[Nessuno] o Nessuno o [1,2,3] o [Nessuno, 3] ma non []	No

Rilevamento dei parametri per WDL

Nel motore di workflow WDL, la logica di analisi si basa sui seguenti presupposti:

- Tutti i tipi nullable supportati sono contrassegnati come parametri di input opzionali.
- Per i tipi supportati che non possono essere annullati:
  - Qualsiasi variabile di input con assegnazione di valori letterali o espressioni è contrassegnata come parametri opzionali. Per esempio:

```
 Int x = 2
Float f0 = 1.0 + f1
```

  - Se non sono stati assegnati valori o espressioni ai parametri di input, questi verranno contrassegnati come parametri obbligatori.
- Le descrizioni vengono estratte dalla definizione `parameter_meta` del `main` flusso di lavoro. Se non `parameter_meta` è specificato, la descrizione sarà vuota (una stringa vuota). Per ulteriori informazioni, vedete la specifica WDL per i metadati dei parametri.

Le tabelle seguenti mostrano esempi di interpolazione WDL. Per ogni esempio, il nome del parametro è. x Se il parametro è obbligatorio, è necessario fornire un valore per il parametro. Se il parametro è facoltativo, non è necessario fornire un valore.

Questa tabella mostra esempi di interpolazione WDL per tipi primitivi.

Input	Esempio di input/output	Richiesto
Int x	1 o 2 o...	Sì
Int x = 2	2	No
Intx = 1+2	3	No
Int x = y+z	y+z	No
Int? x	Nessuno o 1 o 2 o...	Sì
Int? x = 2	Nessuno o 2	No
Int? x = 1+2	Nessuno o 3	No
Int? x = y+z	Nessuno o y+z	No

La tabella seguente mostra esempi di interpolazione WDL per tipi complessi. Un tipo complesso è una raccolta di tipi primitivi.

Input	Esempio di input/output	Richiesto		
Array [Int] x	[1,2,3] o []	Sì		
Matrice [Int] + x	[1], ma non []	Sì		
Array [Int]? x	Nessuno o [] o [1,2,3]	No		
Array [Int?] x	[] o [Nessuno, 3, Nessuno]	Sì		
Array [Int?] =? x	[Nessuno] o Nessuno o [1,2,3] o [Nessuno, 3] ma non []	No		
Esempio di struttura {String a, Int y} più avanti negli input: Sample MySample	`String a = mySample.a Int y = mySample.y`	Sì		
Esempio di struttura {String a, Int y} più avanti negli input: Sample? MySample	`if (defined( mySample)) { String a = mySample.a Int y = mySample.y }`	No		

Rilevamento dei parametri per Nextflow

Per Nextflow, HealthOmics genera il modello di parametro analizzando il file. `nextflow_schema.json` Se la definizione del flusso di lavoro non include un file di schema, HealthOmics analizza il file di definizione del flusso di lavoro principale.

Argomenti

Analisi del file di schema

Affinché l'analisi funzioni correttamente, assicurati che il file di schema soddisfi i seguenti requisiti:

- Il file di schema è denominato `nextflow_schema.json` e si trova nella stessa directory del file di workflow principale.
- Il file di schema è JSON valido come definito in uno dei seguenti schemi:
  - schema json. org/draft/2020-12/schema.
  - schema json. org/draft-07/schema.

HealthOmics analizza il `nextflow_schema.json` file per generare il modello di parametro:

- Estrae tutto ciò properties che è definito nello schema.
- Include la proprietà description se disponibile per la proprietà.
- Identifica se ogni parametro è facoltativo o obbligatorio, in base al required campo della proprietà.

L'esempio seguente mostra un file di definizione e il file dei parametri generato.

```
{
    "$schema": "https://json-schema.org/draft/2020-12/schema",
    "type": "object",
    "$defs": {
        "input_options": {
            "title": "Input options",
```

```
            "type": "object",
            "required": ["input_file"],
            "properties": {
                "input_file": {
                    "type": "string",
                    "format": "file-path",
                    "pattern": "^s3://[a-z0-9.-]{3,63}(?:/\\S*)?$",
                    "description": "description for input_file"
                },
                "input_num": {
                    "type": "integer",
                    "default": 42,
                    "description": "description for input_num"
                }
            }
        },
        "output_options": {
            "title": "Output options",
            "type": "object",
            "required": ["output_dir"],
            "properties": {
                "output_dir": {
                    "type": "string",
                    "format": "file-path",
                    "description": "description for output_dir",
                }
            }
        }
    },
    "properties": {
        "ungrouped_input_bool": {
            "type": "boolean",
            "default": true
        }
    },
    "required": ["ungrouped_input_bool"],
    "allOf": [
        { "$ref": "#/$defs/input_options" },
        { "$ref": "#/$defs/output_options" }
    ]
}
```

Il modello di parametro generato:

```
{
    "input_file": {
        "description": "description for input_file",
        "optional": False
    },
    "input_num": {
        "description": "description for input_num",
        "optional": True
    },
    "output_dir": {
        "description": "description for output_dir",
        "optional": False
    },
    "ungrouped_input_bool": {
        "description": None,
        "optional": False
    }
}
```

Analisi del file principale

Se la definizione del flusso di lavoro non include un `nextflow_schema.json` file, HealthOmics analizza il file di definizione del flusso di lavoro principale.

HealthOmics analizza le `params` espressioni presenti nel file di definizione del flusso di lavoro principale e nel `nextflow.config` file. Tutte le `params` versioni con valori predefiniti sono contrassegnate come opzionali.

Affinché l'analisi funzioni correttamente, tieni presente i seguenti requisiti:

- HealthOmics analizza solo il file di definizione del flusso di lavoro principale. Per garantire che tutti i parametri vengano acquisiti, ti consigliamo di collegarlo a tutti params i sottomoduli e ai flussi di lavoro importati.
- Il file di configurazione è facoltativo. Se ne definite uno, assegnategli un nome `nextflow.config` e inseritelo nella stessa directory del file di definizione del flusso di lavoro principale.

L'esempio seguente mostra un file di definizione e il modello di parametri generato.

```
params.input_file = "default.txt"
params.threads = 4
```

```
params.memory = "8GB"

workflow {
    if (params.version) {
        println "Using version: ${params.version}"
    }
}
```

Il modello di parametro generato:

```
{
    "input_file": {
        "description": None,
        "optional": True
    },
    "threads": {
        "description": None,
        "optional": True
    },
    "memory": {
        "description": None,
        "optional": True
    },
    "version": {
        "description": None,
        "optional": False
    }
}
```

Per i valori predefiniti definiti in nextflow.config, HealthOmics raccoglie le `params` assegnazioni e i parametri dichiarati all'interno`params {}`, come illustrato nell'esempio seguente. Nelle istruzioni di assegnazione, `params` deve apparire nella parte sinistra dell'istruzione.

```
params.alpha = "alpha"
params.beta = "beta"

params {
    gamma = "gamma"
    delta = "delta"
}

env {
```

```
   // ignored, as this assignment isn't in the params block
   VERSION = "TEST"
}

// ignored, as params is not on the left side
interpolated_image = "${params.cli_image}"
```

Il modello di parametro generato:

```
{
    // other params in your main workflow defintion
    "alpha": {
        "description": None,
        "optional": True
    },
    "beta": {
        "description": None,
        "optional": True
    },
    "gamma": {
        "description": None,
        "optional": True
    },
    "delta": {
        "description": None,
        "optional": True
    }
}
```

Parametri annidati

Entrambi `nextflow.config` consentono `nextflow_schema.json` parametri annidati. Tuttavia, il modello di HealthOmics parametri richiede solo i parametri di primo livello. Se il flusso di lavoro utilizza un parametro annidato, è necessario fornire un oggetto JSON come input per quel parametro.

Parametri annidati nei file di schema

HealthOmics salti annidati params durante l'analisi di un file. `nextflow_schema.json` Ad esempio, se si definisce il seguente file: `nextflow_schema.json`

```
{
    "properties": {
```

```
        "input": {
            "properties": {
                "input_file": { ... },
                "input_num": { ... }
            }
        },
        "input_bool": { ... }
    }
}
```

HealthOmics ignora `input_file` e `input_num` quando genera il modello di parametro:

```
{
    "input": {
        "description": None,
        "optional": True
    },
    "input_bool": {
        "description": None,
        "optional": True
    }
}
```

Quando si esegue questo flusso di lavoro, HealthOmics si aspetta un `input.json` file simile al seguente:

```
{
   "input": {
       "input_file": "s3://bucket/obj",
       "input_num": 2
   },
   "input_bool": false
}
```

Parametri annidati nei file di configurazione

HealthOmics non raccoglie i dati annidati params in un `nextflow.config` file e li salta durante l'analisi. Ad esempio, se definisci il seguente file: `nextflow.config`

```
params.alpha = "alpha"
  params.nested.beta = "beta"
```

```
  params {
      gamma = "gamma"
      group {
          delta = "delta"
      }
  }
```

HealthOmics ignora `params.nested.beta` e `params.group.delta` quando genera il modello di parametro:

```
{
    "alpha": {
        "description": None,
        "optional": True
    },
    "gamma": {
        "description": None,
        "optional": True
    }
}
```

Esempi di interpolazione Nextflow

La tabella seguente mostra esempi di interpolazione Nextflow per i parametri nel file principale.

Parametri	Richiesto
params.input_file	Sì
params.input_file = "s3://bucket/data.json»	No
params.nested.input_file	N/D
params.nested.input_file = "s3://bucket/data.json»	N/D

La tabella seguente mostra esempi di interpolazione Nextflow per i parametri nel file. `nextflow.config`

Parametri	Richiesto
`params.input_file = "s3://bucket/data.json"`	No
`params { input_file = "s3://bucket/data.json" }`	No
`params { nested { input_file = "s3://bucket/data.json" } }`	N/D
`input_file = params.input_file`	N/D

Immagini di container per flussi di lavoro privati

HealthOmics supporta immagini di container ospitate in repository privati di Amazon ECR. Puoi creare immagini di contenitori e caricarle nel repository privato. Puoi anche utilizzare il tuo registro privato Amazon ECR come cache pull through per sincronizzare il contenuto dei registri upstream.

Il tuo repository Amazon ECR deve risiedere nella stessa AWS regione dell'account che chiama il servizio. L'immagine del contenitore Account AWS può essere di proprietà di un altro soggetto, a condizione che l'archivio delle immagini di origine fornisca le autorizzazioni appropriate. Per ulteriori informazioni, consulta Politiche per l'accesso ad Amazon ECR su più account.

Ti consigliamo di definire l'immagine del contenitore Amazon ECR URIs come parametri del flusso di lavoro in modo che l'accesso possa essere verificato prima dell'inizio dell'esecuzione. Inoltre, semplifica l'esecuzione di un flusso di lavoro in una nuova regione modificando il parametro Region.

> **Note**
>
> HealthOmics non supporta i contenitori ARM e non supporta l'accesso agli archivi pubblici.

Per informazioni sulla configurazione delle autorizzazioni IAM per l'accesso HealthOmics ad Amazon ECR, consulta. HealthOmics Autorizzazioni per le risorse

Argomenti

- Sincronizzazione con registri di container di terze parti
- Considerazioni generali per le immagini dei container Amazon ECR
- Variabili di ambiente per i flussi HealthOmics di lavoro
- Utilizzo di Java nelle immagini dei container Amazon ECR
- Aggiungi input di attività a un'immagine del contenitore Amazon ECR

Sincronizzazione con registri di container di terze parti

Puoi utilizzare le regole pull through cache di Amazon ECR per sincronizzare i repository in un registro upstream supportato con i tuoi repository privati Amazon ECR. Per ulteriori informazioni, consulta Sincronizzare un registro upstream nella Amazon ECR User Guide.

La cache pull through crea automaticamente l'archivio di immagini nel registro privato al momento della creazione della cache e si sincronizza automaticamente con l'immagine memorizzata nella cache in caso di modifiche all'immagine upstream.

HealthOmics supporta la cache pull through per i seguenti registri upstream:

- Amazon ECR Public
- Registro delle immagini del contenitore Kubernetes
- Quay
- Docker Hub
- Microsoft Azure Container Registry
- GitHub Registro dei contenitori
- GitLab Registro dei contenitori

HealthOmics non supporta la cache pull through per un repository privato Amazon ECR upstream.

I vantaggi dell'utilizzo della cache pull through di Amazon ECR includono:

1. Evita di dover migrare manualmente le immagini dei container su Amazon ECR o sincronizzare gli aggiornamenti dal repository di terze parti.
2. I flussi di lavoro accedono alle immagini sincronizzate dei contenitori nel tuo repository privato, che è più affidabile rispetto al download di contenuti in fase di esecuzione da un registro pubblico.
3. Poiché le cache pull through di Amazon ECR utilizzano una struttura URI prevedibile, il HealthOmics servizio può mappare automaticamente l'URI privato di Amazon ECR con l'URI del registro upstream. Non è necessario aggiornare e sostituire i valori URI nella definizione del flusso di lavoro.

Argomenti

Configurazione della cache pull through

Amazon ECR fornisce un registro per ogni Account AWS regione. Assicurati di creare la configurazione Amazon ECR nella stessa regione in cui intendi eseguire il flusso di lavoro.

Le seguenti sezioni descrivono le attività di configurazione per il pull through cache.

Attività di configurazione

Crea una regola pull through cache

Crea una regola pull through cache di Amazon ECR per ogni registro upstream contenente immagini che desideri memorizzare nella cache. Una regola specifica una mappatura tra un registro upstream e l'archivio privato Amazon ECR.

Per un registro upstream che richiede l'autenticazione, fornisci le tue credenziali utilizzando AWS Secrets Manager.

> **Note**
>
> Non modificare una regola di pull through cache mentre un'esecuzione attiva utilizza il repository privato. L'esecuzione potrebbe fallire o, cosa ancora più importante, far sì che la pipeline utilizzi immagini inaspettate.

Per ulteriori informazioni, consulta Creazione di una regola pull through cache nella Amazon Elastic Container Registry User Guide.

Crea una regola pull through cache utilizzando la console

Per configurare la cache pull through, segui questi passaggi utilizzando la console Amazon ECR:

1. Apri la console Amazon ECR: https://console.aws.amazon.com /ecr
2. Dal menu a sinistra, in Registro privato, espandi Caratteristiche e impostazioni, quindi scegli Pull through cache.
3. Dalla pagina Pull through cache, scegli Aggiungi regola.
4. Nel pannello del registro Upstream, scegli il registro upstream da sincronizzare con il registro privato, quindi scegli Avanti.
5. Se il registro upstream richiede l'autenticazione, la console apre una nuova pagina in cui specifichi il segreto SageMaker AI che contiene le tue credenziali. Scegli Next (Successivo).
6. In Specificate namespaces, nel pannello Cache namespace, scegliete se creare i repository privati utilizzando un prefisso di repository specifico o senza prefisso. Se scegliete di utilizzare un prefisso, specificate il nome del prefisso nel prefisso del repository Cache.
7. Nel pannello dello spazio dei nomi Upstream, scegliete se estrarre dai repository upstream utilizzando un prefisso di repository specifico o senza prefisso. Se scegliete di utilizzare un prefisso, specificate il nome del prefisso nel prefisso del repository Upstream.

   Il pannello di esempio Namespace mostra un esempio di pull request, un URL upstream e l'URL del cache repository che viene creato.
8. Scegli Next (Successivo).
9. Rivedi la configurazione e scegli Crea per creare la regola.

Per ulteriori informazioni, consulta Creare una regola pull through cache (AWS Management Console).

Crea una regola pull through cache utilizzando la CLI

Usa il create-pull-through-cache-rule comando Amazon ECR per creare una regola pull through cache. Per i registri upstream che richiedono l'autenticazione, memorizzate le credenziali in un segreto di Secrets Manager.

Le sezioni seguenti forniscono esempi per ogni registro upstream supportato.

Per Amazon ECR Public

L'esempio seguente crea una regola di cache pull-through per il registro pubblico di Amazon ECR. Specifica un prefisso di `ecr-public`, che fa sì che ciascun repository creato utilizzando la regola della cache pull-through abbia lo schema di denominazione di `ecr-public/`*`upstream-repository-name`*.

```
aws ecr create-pull-through-cache-rule \
     --ecr-repository-prefix ecr-public \
     --upstream-registry-url public.ecr.aws \
     --region us-east-1
```

Per Kubernetes Container Registry

L'esempio seguente crea una regola di cache pull through per il registro pubblico Kubernetes. Specifica un prefisso di `kubernetes`, che fa sì che ciascun repository creato utilizzando la regola della cache pull-through abbia lo schema di denominazione di `kubernetes/`*`upstream-repository-name`*.

```
aws ecr create-pull-through-cache-rule \
     --ecr-repository-prefix kubernetes \
     --upstream-registry-url registry.k8s.io \
     --region us-east-1
```

Per Quay

L'esempio seguente crea una regola di cache pull-through per il registro pubblico Quay. Specifica un prefisso di `quay`, che fa sì che ciascun repository creato utilizzando la regola della cache pull-through abbia lo schema di denominazione di `quay/`*`upstream-repository-name`*.

```
aws ecr create-pull-through-cache-rule \
     --ecr-repository-prefix quay \
     --upstream-registry-url quay.io \
     --region us-east-1
```

Per Docker Hub

L'esempio seguente crea una regola di cache pull-through per il registro Docker Hub. Specifica un prefisso di `docker-hub`, che fa sì che ciascun repository creato utilizzando la regola della cache pull-through abbia lo schema di denominazione di `docker-hub/`*`upstream-repository-name`*. Devi specificare nella sua interezza il nome della risorsa Amazon (ARN) del segreto contenente le credenziali di Docker Hub.

```
aws ecr create-pull-through-cache-rule \
     --ecr-repository-prefix docker-hub \
     --upstream-registry-url registry-1.docker.io \
     --credential-arn arn:aws:secretsmanager:us-east-1:111122223333:secret:ecr-
pullthroughcache/example1234 \
     --region us-east-1
```

Per Container Registry GitHub

L'esempio seguente crea una regola pull through cache per il GitHub Container Registry. Specifica un prefisso di `github`, che fa sì che ciascun repository creato utilizzando la regola della cache pull-through abbia lo schema di denominazione di `github/`*`upstream-repository-name`*.
È necessario specificare l'Amazon Resource Name (ARN) completo del segreto contenente le credenziali del GitHub Container Registry.

```
aws ecr create-pull-through-cache-rule \
     --ecr-repository-prefix github \
     --upstream-registry-url ghcr.io \
     --credential-arn arn:aws:secretsmanager:us-east-1:111122223333:secret:ecr-
pullthroughcache/example1234 \
     --region us-east-1
```

Per Microsoft Azure Container Registry

L'esempio seguente crea una regola pull through cache per il Microsoft Azure Container Registry. Specifica un prefisso di `azure`, che fa sì che ciascun repository creato utilizzando la regola della

cache pull-through abbia lo schema di denominazione di `azure/`*`upstream-repository-name`*. Devi specificare nella sua interezza il nome della risorsa Amazon (ARN) del segreto contenente le credenziali di Microsoft Azure Container Registry.

```
aws ecr create-pull-through-cache-rule \
     --ecr-repository-prefix azure \
     --upstream-registry-url myregistry.azurecr.io \
     --credential-arn arn:aws:secretsmanager:us-east-1:111122223333:secret:ecr-
pullthroughcache/example1234 \
     --region us-east-1
```

Per GitLab Container Registry

L'esempio seguente crea una regola pull through cache per il GitLab Container Registry. Specifica un prefisso di `gitlab`, che fa sì che ciascun repository creato utilizzando la regola della cache pull-through abbia lo schema di denominazione di `gitlab/`*`upstream-repository-name`*.
È necessario specificare l'Amazon Resource Name (ARN) completo del segreto contenente le credenziali del GitLab Container Registry.

```
aws ecr create-pull-through-cache-rule \
     --ecr-repository-prefix gitlab \
     --upstream-registry-url registry.gitlab.com \
     --credential-arn arn:aws:secretsmanager:us-east-1:111122223333:secret:ecr-
pullthroughcache/example1234 \
     --region us-east-1
```

Per ulteriori informazioni, consulta Create a pull through cache rule (CLI) nella Amazon ECR User Guide.

È possibile utilizzare il comando get-run-task CLI per recuperare informazioni sull'immagine del contenitore utilizzata per un'attività specifica:

```
aws omics get-run-task --id 1234567 --task-id <task_id>
```

L'output include le seguenti informazioni sull'immagine del contenitore:

```
"imageDetails": {
   "image": "string",
   "imageDigest": "string",
```

```
    "sourceImage": "string",
          ...
}
```

Autorizzazioni di registro per il registro upstream

Utilizza le autorizzazioni di registro HealthOmics per consentire l'utilizzo della cache pull through e per inserire le immagini del contenitore nel registro privato di Amazon ECR. Aggiungi una policy del registro Amazon ECR al registro che fornisce i contenitori utilizzati nelle esecuzioni.

La seguente policy concede al HealthOmics servizio l'autorizzazione a creare repository con i prefissi pull through cache specificati e ad avviare recuperi upstream in questi repository.

1. Dalla console Amazon ECR, apri il menu a sinistra, in Registro privato, espandi le autorizzazioni del registro, quindi scegli Genera dichiarazione.
2. In alto a destra, scegli JSON. Inserisci una politica simile alla seguente:

   JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Sid": "AllowPTCinRegPermissions",
            "Effect": "Allow",
            "Principal": {
                "Service": "omics.amazonaws.com"
            },
            "Action": [
                "ecr:CreateRepository",
                "ecr:BatchImportUpstreamImage"
            ],
            "Resource": [
                "arn:aws:ecr:us-east-1:123456789012:repository/ecr-public/*",
                "arn:aws:ecr:us-east-1:123456789012:repository/docker-hub/*"
            ]
        }
    ]
}
```

Modelli per la creazione di repository

Per utilizzare il pull through caching in HealthOmics, il repository Amazon ECR deve disporre di un modello di creazione del repository. Il modello definisce le impostazioni di configurazione per quando tu o Amazon ECR create un repository privato per un registro upstream.

Ogni modello contiene un prefisso dello spazio dei nomi del repository, che Amazon ECR utilizza per abbinare i nuovi repository a un modello specifico. I modelli specificano la configurazione per tutte le impostazioni del repository, comprese le politiche di accesso basate sulle risorse, l'immutabilità dei tag, la crittografia e le politiche del ciclo di vita.

Per ulteriori informazioni, consulta Modelli di creazione di repository nella Amazon Elastic Container Registry User Guide.

Come creare un modello per la creazione di un repository:

1. Dalla console Amazon ECR, apri il menu a sinistra, in Registro privato, espandi Caratteristiche e impostazioni, quindi scegli Modelli di creazione di repository.
2. Scegli Crea modello.
3. Nei dettagli del modello, scegli Pull through cache.
4. Scegli se applicare questo modello a un prefisso specifico o a tutti i repository che non corrispondono a un altro modello.

   Se scegli Un prefisso specifico, inserisci il valore del prefisso dello spazio dei nomi in Prefisso. Avete specificato questo prefisso quando avete creato la regola PTC.
5. Scegli Next (Successivo).
6. Nella pagina Aggiungi configurazione per la creazione del repository, immettete le autorizzazioni per la creazione del repository. Utilizza una delle istruzioni politiche di esempio o inseriscine una simile all'esempio seguente:

   JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Sid": "PTCRepoCreationTemplate",
            "Effect": "Allow",
            "Principal": {
```

```
                "Service": "omics.amazonaws.com"
            },
            "Action": [
                "ecr:BatchGetImage",
                "ecr:GetDownloadUrlForLayer"
            ],
            "Resource": "*"
        }
    ]
}
```

7. Facoltativamente, puoi aggiungere impostazioni del repository come i criteri del ciclo di vita e i tag. Amazon ECR applica queste regole a tutte le immagini di container create per la cache pull through che utilizzano il prefisso specificato.
8. Scegli Next (Successivo).
9. Controlla la configurazione e scegli Avanti.

Creazione del flusso di lavoro

Quando crei un nuovo flusso di lavoro o una nuova versione del flusso di lavoro, rivedi i mapping del registro e aggiornali se necessario. Per informazioni dettagliate, vedi Creare un flusso di lavoro privato.

Mappature del registro

Definisci le mappature del registro da mappare tra i prefissi nel tuo registro Amazon ECR privato e i nomi del registro upstream.

Per ulteriori informazioni sulle mappature del registro Amazon ECR, consulta Creazione di una regola pull through cache in Amazon ECR.

L'esempio seguente mostra le mappature del registro su Docker Hub, Quay e Amazon ECR Public.

```
{
    "registryMappings": [
        {
            "upstreamRegistryUrl": "registry-1.docker.io",
            "ecrRepositoryPrefix": "docker-hub"
        },
        {
            "upstreamRegistryUrl": "quay.io",
            "ecrRepositoryPrefix": "quay"
```

```
        },
        {
            "upstreamRegistryUrl": "public.ecr.aws",
            "ecrRepositoryPrefix": "ecr-public"
        }
    ]
}
```

Mappature delle immagini

Definisci le mappature delle immagini da mappare tra i nomi delle immagini definiti nei flussi di lavoro privati di Amazon ECR e i nomi delle immagini nel registro upstream.

Puoi utilizzare le mappature delle immagini con registri che supportano la cache pull through. È inoltre possibile utilizzare le mappature delle immagini con registri upstream che non supportano il pull through cache. HealthOmics È necessario sincronizzare manualmente il registro upstream con il repository privato.

Per ulteriori informazioni sulle mappature delle immagini di Amazon ECR, consulta Creazione di una regola pull through cache in Amazon ECR.

L'esempio seguente mostra le mappature da immagini private di Amazon ECR a un'immagine genomica pubblica e all'ultima immagine di Ubuntu.

```
{
    "imageMappings": [
        {
            "sourceImage": "public.ecr.aws/aws-genomics/broadinstitute/gatk:4.6.0.2",
            "destinationImage": "123456789012.dkr.ecr.us-east-1.amazonaws.com/
broadinstitute/gatk:4.6.0.2"
        },
        {
            "sourceImage": "ubuntu:latest",
            "destinationImage": "123456789012.dkr.ecr.us-east-1.amazonaws.com/custom/
ubuntu:latest",
        }
    ]
}
```

Considerazioni generali per le immagini dei container Amazon ECR

- Architecture

HealthOmics supporta contenitori x86_64. Se il computer locale è basato su ARM, come Apple Mac, usa un comando come il seguente per creare un'immagine del contenitore x86_64:

```
docker build --platform amd64 -t my_tool:latest .
```

- Entrypoint e shell

  HealthOmics i motori di workflow iniettano gli script di bash come comando che sostituiscono le immagini dei contenitori utilizzate dalle attività del flusso di lavoro. Pertanto, le immagini dei contenitori dovrebbero essere create senza un ENTRYPOINT specificato in modo tale che una shell bash sia l'impostazione predefinita.

- Percorsi montati

  Un filesystem condiviso viene montato sulle attività del contenitore in /tmp. Tutti i dati o gli strumenti incorporati nell'immagine del contenitore in questa posizione verranno sovrascritti.

  La definizione del flusso di lavoro è disponibile per le attività tramite un montaggio in sola lettura in /mnt/workflow.

- Dimensione dell'immagine

  Vedi per le dimensioni massime delle immagini dei contenitori. HealthOmics quote a dimensione fissa per il flusso di lavoro

Variabili di ambiente per i flussi HealthOmics di lavoro

HealthOmics fornisce variabili di ambiente che contengono informazioni sul flusso di lavoro in esecuzione nel contenitore. È possibile utilizzare i valori di queste variabili nella logica delle attività del flusso di lavoro.

Tutte le variabili HealthOmics del flusso di lavoro iniziano con il `AWS_WORKFLOW_` prefisso. Questo prefisso è un prefisso di variabile di ambiente protetto. Non utilizzate questo prefisso per le vostre variabili nei contenitori del flusso di lavoro.

HealthOmics fornisce le seguenti variabili di ambiente di flusso di lavoro:

AWS_REGION

Questa variabile è la regione in cui è in esecuzione il contenitore.

AWS_WORKFLOW_ESEGUI

Questa variabile è il nome dell'esecuzione corrente.

AWS_WORKFLOW_RUN_ID

Questa variabile è l'identificatore di esecuzione della corsa corrente.

AWS_WORKFLOW_RUN_UUID

Questa variabile è l'UUID di esecuzione dell'esecuzione corrente.

AWS_WORKFLOW_ATTIVITÀ

Questa variabile è il nome dell'attività corrente.

AWS_WORKFLOW_TASK_ID

Questa variabile è l'identificatore dell'attività corrente.

AWS_WORKFLOW_TASK_UUID

Questa variabile è l'UUID dell'attività corrente.

L'esempio seguente mostra i valori tipici per ogni variabile di ambiente:

```
AWS Region: us-east-1
Workflow Run: arn:aws:omics:us-east-1:123456789012:run/6470304
Workflow Run ID: 6470304
Workflow Run UUID: f4d9ed47-192e-760e-f3a8-13afedbd4937
Workflow Task: arn:aws:omics:us-east-1:123456789012:task/4192063
Workflow Task ID: 4192063
Workflow Task UUID: f0c9ed49-652c-4a38-7646-60ad835e0a2e
```

Utilizzo di Java nelle immagini dei container Amazon ECR

Se un'attività di workflow utilizza un'applicazione Java come GATK, considera i seguenti requisiti di memoria per il contenitore:

- Le applicazioni Java utilizzano la memoria stack e la memoria heap. Per impostazione predefinita, la memoria heap massima è una percentuale della memoria totale disponibile nel contenitore. Questa impostazione predefinita dipende dalla distribuzione JVM specifica e dalla versione JVM, quindi consulta la documentazione pertinente per la tua JVM o imposta esplicitamente il massimo della memoria heap utilizzando le opzioni della riga di comando Java (come `-Xmx`).

- Non impostate che la memoria heap massima sia pari al 100% dell'allocazione di memoria del contenitore, poiché anche lo stack JVM richiede memoria. La memoria è richiesta anche per il Garbage Collector JVM e per qualsiasi altro processo del sistema operativo in esecuzione nel contenitore.
- Alcune applicazioni Java, come GATK, possono utilizzare invocazioni di metodi nativi o altre ottimizzazioni come i file di mappatura della memoria. Queste tecniche richiedono allocazioni di memoria eseguite «off heap», che non sono controllate dal parametro JVM maximum heap.

  Se sai (o sospetti) che la tua applicazione Java alloca memoria off-heap, assicurati che l'allocazione della memoria delle attività includa i requisiti di memoria off-heap.

  Se queste allocazioni off-heap causano l'esaurimento della memoria nel contenitore, in genere non viene visualizzato un OutOfMemory errore Java, perché la JVM non controlla questa memoria.

Aggiungi input di attività a un'immagine del contenitore Amazon ECR

Aggiungi tutti gli eseguibili, le librerie e gli script necessari per eseguire un'attività di workflow nell'immagine Amazon ECR utilizzata per eseguire l'attività.

È consigliabile evitare l'uso di script, file binari e librerie esterni all'immagine del contenitore di attività. Ciò è particolarmente importante quando si utilizzano `nf-core` flussi di lavoro che utilizzano una `bin` directory come parte del pacchetto workflow. Sebbene questa directory sia disponibile per l'attività del flusso di lavoro, viene montata come directory di sola lettura. Le risorse richieste in questa directory devono essere copiate nell'immagine dell'attività e rese disponibili in fase di esecuzione o durante la creazione dell'immagine del contenitore utilizzata per l'attività.

Vedi HealthOmics quote a dimensione fissa per il flusso di lavoro la dimensione massima dell'immagine del contenitore HealthOmics supportata.

HealthOmics File README del flusso di lavoro

Puoi caricare un file README.md contenente istruzioni, diagrammi e informazioni essenziali per il tuo flusso di lavoro. Ogni versione del flusso di lavoro supporta un file README, che è possibile aggiornare in qualsiasi momento.

I requisiti README includono:

- Il file README deve essere in formato markdown (.md)
- Dimensione massima del file: 500 KiB

Argomenti

Usa un file README esistente

READMEs i file esportati dai repository Git contengono collegamenti relativi che in genere non funzionano al di fuori del repository. HealthOmics L'integrazione con Git li converte automaticamente in link assoluti per un corretto rendering nella console, eliminando la necessità di aggiornamenti manuali degli URL.

Per l' READMEs importazione da Amazon S3 o da unità locali, le immagini e i link devono essere pubblici URLs o avere i relativi percorsi aggiornati per un rendering corretto.

> **Note**
>
> Le immagini devono essere ospitate pubblicamente per essere visualizzate nella HealthOmics console. Le immagini archiviate nei GitHub Enterprise Server nostri GitLab Self-Managed archivi non possono essere renderizzate.

Condizioni di rendering

La HealthOmics console interpola immagini e collegamenti accessibili al pubblico utilizzando percorsi assoluti. Per eseguire il rendering URLs da archivi privati, l'utente deve avere accesso al repository. I nostri GitHub Enterprise Server GitLab Self-Managed repository, che utilizzano domini personalizzati, HealthOmics non possono risolvere collegamenti relativi o eseguire il rendering di immagini archiviate in questi archivi privati.

La tabella seguente mostra gli elementi markdown supportati dalla visualizzazione README della AWS console.

Elemento	AWS console
Avvisi	Sì, ma senza icone
Distintivi	Sì

Elemento	AWS console
Formattazione di base del testo	Sì
Blocchi di codice	Sì, ma non dispone della funzionalità di evidenziazione della sintassi e del pulsante di copia
Sezioni pieghevoli	Sì
Titoli	Sì
Formati di immagine	Sì
Immagine (cliccabile)	Sì
Interruzioni di riga	Sì
Diagramma a sirena	Solo può aprire il grafico, spostare la posizione del grafico e copiare il codice
Preventivi	Sì
Pedice e apice	Sì
Tabelle	Sì, ma non supporta l'allineamento del testo
Allineamento del testo	Sì

Utilizzo di immagini e link URLs

A seconda del provider di origine, struttura la base URLs per pagine e immagini nei seguenti formati.

- `{username}`: il nome utente in cui è ospitato il repository.
- `{repo}`: il nome del repository.
- `{ref}`: il riferimento alla fonte (branch, tag e commit ID).
- `{path}`: il percorso del file alla pagina o all'immagine nel repository.

Provider di origine	URL della pagina	URL dell'immagine

GitHub	`https://github.com/{username}/{repo}/blob/{ref}/{path}`	`https://github.com/{username}/{repo}/blob/{ref}/{path}?raw=true`
		
		`https://raw.githubusercontent.com/{username}/{repo}/{ref}/{path}`

GitLab	`https://gitlab.com/{username}/{repo}/-/blob/{ref}/{path}`	`https://gitlab.com/{username}/{repo}/-/raw/{ref}/{path}`
Bitbucket	`https://bitbucket.org/{username}/{repo}/src/{ref}/{path}`	`https://bitbucket.org/{username}/{repo}/raw/{ref}/{path}`

GitHubGitLab, e Bitbucket supporta sia la pagina URLs che l'immagine che rimandano a un archivio pubblico. La tabella seguente mostra il supporto di ciascun fornitore di sorgenti per il rendering di immagini e collegamenti URLs per repository privati.

Supporto per repository privati		
Provider di origine	URL della pagina	URL dell'immagine
GitHub	Solo con accesso al repository	No
GitLab	Solo con accesso al repository	No
Bitbucket	Solo con accesso al repository	No

Richiesta di licenze Sentieon per flussi di lavoro privati

Se il tuo flusso di lavoro privato utilizza il software Sentieon, hai bisogno di una licenza Senieon. Segui questi passaggi per richiedere e configurare una licenza per il software Sentieon:

- Richiedi una licenza Sentieon
  - Invia un'e-mail al gruppo di supporto Sentieon (support@sentieon.com) per richiedere una licenza software.
    - Fornisci il tuo ID utente AWS Canonical nell'e-mail.
    - Trova il tuo ID utente AWS Canonical seguendo queste istruzioni.
- Aggiorna il tuo ruolo HealthOmics di servizio per concedergli l'accesso al proxy del server di licenza Sentieon e al bucket Sentieon Omics nella tua regione. L'esempio seguente concede l'accesso a. `us-east-1` Se necessario, sostituisci questo testo con la tua regione.

  JSON

```
{
          "Version":"2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "s3:GetObjectAcl",
                "s3:GetObject"
            ],
            "Resource": [
                "arn:aws:s3:::omics-ap-us-east-1/*",
                "arn:aws:s3:::sentieon-omics-license-us-east-1/*"
            ]
        }
    ]
 }
```

- Genera una AWS richiesta di supporto per accedere al proxy del server di licenza Sentieon.
  - Per creare una richiesta di supporto, accedi a support.console.aws.amazon.com.
  - Fornisci la tua e la regione nella richiesta di assistenza. Account AWS L'account viene aggiunto alla lista consentita per il proxy del server di licenza.
- Crea il tuo flusso di lavoro privato utilizzando il contenitore Sentieon e lo script di licenza Sentieon.

  - Per ulteriori istruzioni sull'uso degli strumenti Sentieon all'interno di flussi di lavoro privati, consulta Sentieon-Amazon-Omics in. GitHub
- La versione del software Sentieon 202112.07 e successive supportano il proxy del server di licenza. HealthOmics Per utilizzare le versioni del software Sentieon precedenti alla 202112.07, contatta l'assistenza Sentieon.

Stampanti per flussi di lavoro in HealthOmics

Dopo aver creato un flusso di lavoro, ti consigliamo di eseguire un linter sul flusso di lavoro prima di iniziare la prima esecuzione. Il linter rileva gli errori che possono causare il fallimento dell'esecuzione.

Per WDL, esegue HealthOmics automaticamente un linter quando si crea il flusso di lavoro. L'output del linter è disponibile nel `statusMessage` campo della risposta. get-workflow Utilizzate il seguente comando CLI per recuperare l'output di stato (utilizzate l'ID del flusso di lavoro WDL che avete creato):

```
aws omics get-workflow
   –id 123456
   –query 'statusMessage'
```

HealthOmics fornisce linter che è possibile eseguire sulla definizione del flusso di lavoro prima di creare il flusso di lavoro. Esegui questi linter sulle pipeline esistenti verso cui stai migrando. HealthOmics

- WDL— Un'immagine Amazon ECR pubblica per eseguire un linter WDL.

- Nextflow— Un'immagine Amazon ECR pubblica per eseguire le regole Linter per Nextflow. Puoi accedere al codice sorgente di questo linter da. GitHub
- CWL— non disponibile

HealthOmics operazioni del flusso di lavoro

Per creare un flusso di lavoro privato, devi:

- Workflow definition file:Un file di definizione del flusso di lavoro scritto in WDLNextflow, oCWL. La definizione del flusso di lavoro specifica gli input e gli output per le esecuzioni che utilizzano

il flusso di lavoro. Include inoltre le specifiche per le attività di esecuzione ed esecuzione del flusso di lavoro, inclusi i requisiti di calcolo e memoria. Il file di definizione del flusso di lavoro deve essere in `.zip` formato. Per ulteriori informazioni, vedere File di definizione del flusso di lavoro in HealthOmics.

  - Puoi utilizzare Amazon Q CLI per creare e convalidare i file di definizione del flusso di lavoro in WDL, Nextflow e CWL. Per ulteriori informazioni, consulta le istruzioni di esempio per la CLI di Amazon Q e il tutorial sull'intelligenza artificiale generativa HealthOmics Agentic su. GitHub

- (Optional) Parameter template file:Un file modello di parametro scritto in. JSON Crea il file per definire i parametri di esecuzione o HealthOmics genera automaticamente il modello dei parametri. Per ulteriori informazioni, consultate File modello di parametri per i HealthOmics flussi di lavoro.
- Amazon ECR container images:Crea repository Amazon ECR privati per ogni contenitore utilizzato nel flusso di lavoro. Crea immagini di container per il flusso di lavoro e memorizzale in un repository privato oppure sincronizza il contenuto di un registro upstream supportato con il tuo repository privato ECR.
- (Optional) Sentieon licenses:Richiedi una Sentieon licenza per utilizzare il software in flussi di lavoro privatiSentieon.

Per i file di definizione del flusso di lavoro più grandi di 4 MiB (compressi), scegli una di queste opzioni durante la creazione del flusso di lavoro:

- Carica in una cartella Amazon Simple Storage Service e specifica la posizione.
- Carica su un repository esterno (dimensione massima 1 GiB) e specifica i dettagli del repository.

Dopo aver creato un flusso di lavoro, è possibile aggiornare le seguenti informazioni sul flusso di lavoro con l'operazione: `UpdateWorkflow`

- Nome
- Description
- Tipo di archiviazione predefinito
- Capacità di archiviazione predefinita (con ID del flusso di lavoro)
- File README.md

Per modificare altre informazioni nel flusso di lavoro, crea un nuovo flusso di lavoro o una nuova versione del flusso di lavoro.

Utilizza il controllo delle versioni del flusso di lavoro per organizzare e strutturare i flussi di lavoro. Le versioni consentono inoltre di gestire l'introduzione di aggiornamenti iterativi del flusso di lavoro. Per ulteriori informazioni sulle versioni, consulta Creare una versione del flusso di lavoro.

Argomenti

Creare un flusso di lavoro privato

Crea un flusso di lavoro utilizzando la HealthOmics console, i comandi AWS CLI o uno dei. AWS SDKs

> **Note**
>
> Non includere informazioni di identificazione personale (PII) nei nomi dei flussi di lavoro. Questi nomi sono visibili nei CloudWatch registri.

Quando crei un flusso di lavoro, HealthOmics assegna un identificatore univoco universale (UUID) al flusso di lavoro. L'UUID del flusso di lavoro è un identificatore univoco globale (GUID) unico per i flussi di lavoro e le versioni del flusso di lavoro. Ai fini della provenienza dei dati, si consiglia di utilizzare l'UUID del flusso di lavoro per identificare in modo univoco i flussi di lavoro.

Se le attività del flusso di lavoro utilizzano strumenti esterni (eseguibili, librerie o script), questi strumenti vengono incorporati in un'immagine contenitore. Sono disponibili le seguenti opzioni per ospitare l'immagine del contenitore:

- Ospita l'immagine del contenitore nel registro privato ECR. Prerequisiti per questa opzione:
  - Crea un archivio privato ECR o scegli un repository esistente.
  - Configurare la politica delle risorse ECR come descritto in. Autorizzazioni Amazon ECR
  - Carica l'immagine del contenitore nell'archivio privato.

- Sincronizza l'immagine del contenitore con il contenuto di un registro di terze parti supportato. Prerequisiti per questa opzione:
  - Nel registro privato ECR, configura una regola pull through cache per ogni registro upstream. Per ulteriori informazioni, consulta Mappature delle immagini.
  - Configura la politica delle risorse ECR come descritto in. Autorizzazioni Amazon ECR
  - Crea modelli per la creazione di repository. Il modello definisce le impostazioni per quando Amazon ECR crea l'archivio privato per un registro upstream.
  - Crea mappature di prefissi per rimappare i riferimenti alle immagini dei contenitori nella definizione del flusso di lavoro ai namespace della cache ECR.

Quando si crea un flusso di lavoro, si fornisce una definizione del flusso di lavoro che contiene informazioni sul flusso di lavoro, sulle esecuzioni e sulle attività. HealthOmics può recuperare la definizione del flusso di lavoro come archivio.zip archiviato localmente o in un bucket Amazon S3 o da un repository basato su Git supportato.

Argomenti

- Creare un flusso di lavoro utilizzando la console
- Creazione di un flusso di lavoro utilizzando la CLI
- Creazione di un flusso di lavoro utilizzando un SDK

Creare un flusso di lavoro utilizzando la console

Passaggi per creare un flusso di lavoro

1. Apri la HealthOmics console.
2. Se necessario, apri il riquadro di navigazione a sinistra (≡). Scegli Flussi di lavoro privati.
3. Nella pagina Flussi di lavoro privati, scegli Crea flusso di lavoro.
4. Nella pagina Definisci flusso di lavoro, fornisci le seguenti informazioni:
   1. Nome del flusso di lavoro: un nome distintivo per questo flusso di lavoro. Ti consigliamo di impostare i nomi dei flussi di lavoro per organizzare le esecuzioni nella AWS HealthOmics console e nei CloudWatch registri.
   2. Descrizione (opzionale): una descrizione di questo flusso di lavoro.
5. Nel pannello di definizione del flusso di lavoro, fornite le seguenti informazioni:

1. Lingua del flusso di lavoro (opzionale): seleziona la lingua specifica del flusso di lavoro. Altrimenti, HealthOmics determina la lingua dalla definizione del flusso di lavoro.
2. Per l'origine delle definizioni di Workflow, scegli di importare la cartella delle definizioni da un repository basato su Git, una posizione Amazon S3 o da un'unità locale.

   a. Per l'importazione da un servizio di repository:

   > **Note**
   >
   > HealthOmics supporta repository pubblici e privati perGitHub,,, GitLabBitbucket,GitHub self-managed. GitLab self-managed

   i. Scegli una connessione per connettere AWS le tue risorse al repository esterno. Per creare una connessione, consultaConnect con repository di codice esterni.

   > **Note**
   >
   > I clienti della TLV regione devono creare una connessione nella regione IAD (us-east-1) per creare un flusso di lavoro.

   ii. In Full repository ID, inserisci l'ID del repository come nome utente/nome del repository. Verifica di avere accesso ai file in questo repository.

   iii. In Riferimento alla fonte (opzionale), inserisci un riferimento alla fonte del repository (branch, tag o ID di commit). HealthOmics utilizza il ramo predefinito se non viene specificato alcun riferimento alla fonte.

   iv. In Escludi modelli di file, inserisci i modelli di file per escludere cartelle, file o estensioni specifici. Questo aiuta a gestire le dimensioni dei dati durante l'importazione dei file del repository. Esistono un massimo di 50 pattern e i pattern devono seguire la sintassi del modello glob. Ad esempio:

   A. `tests/`

   B. `*.jpeg`

   C. `large_data.zip`

   b. Per Seleziona la cartella di definizione da S3:

i. Inserisci la posizione Amazon S3 che contiene la cartella di definizione del flusso di lavoro compressa. Il bucket Amazon S3 deve trovarsi nella stessa regione del flusso di lavoro.

ii. Se il tuo account non possiede il bucket Amazon S3, inserisci l'ID dell'account del proprietario del bucket nell'ID dell' AWS account del proprietario del bucket S3. Queste informazioni sono necessarie per verificare la proprietà del HealthOmics bucket.

c. Per Seleziona la cartella di definizione da una fonte locale:

i. Immettete la posizione sull'unità locale della cartella di definizione del flusso di lavoro compressa.

3. Percorso principale del file di definizione del flusso di lavoro (opzionale): immettete il percorso del file dalla cartella o dall'archivio di definizione del flusso di lavoro compresso al file. `main` Questo parametro non è richiesto se è presente un solo file nella cartella di definizione del flusso di lavoro o se il file principale è denominato «principale».

6. Nel pannello del file README (opzionale), selezionate l'origine del file README e fornite le seguenti informazioni:

- Per Importazione da un servizio di repository, nel percorso del file README, inserisci il percorso del file README all'interno del repository.
- Per Seleziona file da S3, nel file README in S3, inserisci l'URI Amazon S3 per il file README.
- Per Seleziona file da una fonte locale: in README - opzionale, scegli Scegli file per selezionare il file markdown (.md) da caricare.

7. Nel pannello di configurazione dell'archiviazione di esecuzione predefinita, fornisci il tipo e la capacità di archiviazione di esecuzione predefiniti per le esecuzioni che utilizzano questo flusso di lavoro:

1. Tipo di archiviazione di esecuzione: scegli se utilizzare l'archiviazione statica o dinamica come impostazione predefinita per l'archiviazione di esecuzione temporanea. L'impostazione predefinita è l'archiviazione statica.
2. Capacità di archiviazione di esecuzione (opzionale): per il tipo di storage di esecuzione statico, è possibile inserire la quantità predefinita di storage di esecuzione richiesta per questo flusso di lavoro. Il valore predefinito per questo parametro è 1200 GiB. È possibile sovrascrivere questi valori predefiniti quando si avvia un'esecuzione.

8. Tag (opzionale): puoi associare fino a 50 tag a questo flusso di lavoro.

9. Scegli Next (Successivo).

10. Nella pagina Aggiungi parametri del flusso di lavoro (opzionale), seleziona l'origine dei parametri:

    1. Per Analizza dal file di definizione del flusso di lavoro, HealthOmics analizzerà automaticamente i parametri del flusso di lavoro dal file di definizione del flusso di lavoro.
    2. Per Fornire un modello di parametro dal repository Git, usa il percorso del file del modello di parametro dal tuo repository.
    3. Per Seleziona il file JSON dalla fonte locale, carica un JSON file da una fonte locale che specifichi i parametri.
    4. In Inserisci manualmente i parametri del flusso di lavoro, inserisci manualmente i nomi e le descrizioni dei parametri.
11. Nel pannello di anteprima dei parametri, è possibile rivedere o modificare i parametri per questa versione del flusso di lavoro. Se ripristinate il JSON file, perderete tutte le modifiche locali apportate.
12. Scegli Next (Successivo).
13. Nella pagina di rimappatura dell'URI del contenitore, nel pannello Regole di mappatura, puoi definire le regole di mappatura URI per il tuo flusso di lavoro.

    Per Origine del file di mappatura, selezionate una delle seguenti opzioni:

    - Nessuna: non è richiesta alcuna regola di mappatura.
    - Seleziona il file JSON da S3: specifica la posizione S3 per il file di mappatura.
    - Seleziona il file JSON da una fonte locale: specifica la posizione del file di mappatura sul tuo dispositivo locale.
    - Inserisci manualmente le mappature: inserisci le mappature del registro e le mappature delle immagini nel pannello Mappature.
14. La console visualizza il pannello Mappature. Se avete scelto un file sorgente di mappatura, la console visualizza i valori del file.

    a. Nelle mappature del registro, è possibile modificare le mappature o aggiungere mappature (massimo 20 mappature del registro).

       Ogni mappatura del registro contiene i seguenti campi:

       - URL del registro upstream: l'URI del registro upstream.
       - Prefisso del repository ECR: il prefisso del repository da utilizzare nell'archivio privato Amazon ECR.

- (Facoltativo) Prefisso del repository upstream: il prefisso del repository nel registro upstream.
- (Facoltativo) ID account ECR: ID account dell'account proprietario dell'immagine del contenitore upstream.

b. Nelle mappature delle immagini, è possibile modificare le mappature delle immagini o aggiungere mappature (massimo 100 mappature di immagini).

Ogni mappatura di immagini contiene i seguenti campi:

- Immagine di origine: specifica l'URI dell'immagine di origine nel registro upstream.
- Immagine di destinazione: specifica l'URI dell'immagine corrispondente nel registro privato di Amazon ECR.

15. Scegli Next (Successivo).
16. Controlla la configurazione del flusso di lavoro, quindi scegli Crea flusso di lavoro.

Creazione di un flusso di lavoro utilizzando la CLI

Se i file del flusso di lavoro e il file del modello dei parametri si trovano sul computer locale, è possibile creare un flusso di lavoro utilizzando il seguente comando CLI.

```
aws omics create-workflow  \
  --name "my_workflow"   \
  --definition-zip fileb://my-definition.zip \
  --parameter-template file://my-parameter-template.json
```

L'`create-workflow`operazione restituisce la seguente risposta:

```
{
  "arn": "arn:aws:omics:us-west-2:....",
  "id": "1234567",
  "status": "CREATING",
  "tags": {
      "resourceArn": "arn:aws:omics:us-west-2:...."
  },
  "uuid": "64c9a39e-8302-cc45-0262-2ea7116d854f"
}
```

Parametri opzionali da utilizzare durante la creazione di un flusso di lavoro

È possibile specificare uno qualsiasi dei parametri opzionali quando si crea un flusso di lavoro. Per i dettagli sulla sintassi, CreateWorkflowconsulta AWS HealthOmics API Reference.

Argomenti

- Specificare la definizione del flusso di lavoro (posizione Amazon S3)
- Usa la definizione del flusso di lavoro da un repository basato su Git
- Specificare un file Readme
- Specificare il file di definizione main
- Specificate il tipo di archiviazione di esecuzione
- Specificare la configurazione della GPU
- Configura i parametri di mappatura pull through cache

Specificare la definizione del flusso di lavoro (posizione Amazon S3)

Se il file di definizione del flusso di lavoro si trova in una cartella Amazon S3, specifica la posizione utilizzando il `definition-uri` parametro, come mostrato nell'esempio seguente. Se il tuo account non possiede il bucket Amazon S3, fornisci l'ID del proprietario. Account AWS

```
aws omics create-workflow  \
  --name Test  \
  --definition-uri s3://omics-bucket/workflow-definition/  \
  --owner-id  123456789012
    ...
```

Usa la definizione del flusso di lavoro da un repository basato su Git

Per utilizzare la definizione del flusso di lavoro da un repository basato su Git supportato, usa il `definition-repository` parametro nella richiesta. Non fornire nessun altro `definition` parametro, poiché una richiesta ha esito negativo se include più di una fonte di input.

Il `definition-respository` parametro contiene i seguenti campi:

- connectionArn— ARN della Code Connection che collega le risorse AWS al repository esterno.
- fullRepositoryId— Inserisci l'ID del repository come. `owner-name/repo-name` Verifica di avere accesso ai file in questo repository.

- sourceReference(Facoltativo): immettete un tipo di riferimento del repository (BRANCH, TAG o COMMIT) e un valore.

  HealthOmics utilizza il commit più recente sul ramo predefinito se non specificate un riferimento alla fonte.

- excludeFilePatterns(Facoltativo): immettete i modelli di file per escludere cartelle, file o estensioni specifiche. Questo aiuta a gestire le dimensioni dei dati durante l'importazione dei file del repository. Fornisci un massimo di 50 pattern. I pattern devono seguire la sintassi del modello glob. Ad esempio:
  - `tests/`
  - `*.jpeg`
  - `large_data.zip`

Quando specifichi la definizione del flusso di lavoro da un repository basato su Git, usa `parameter-template-path` per specificare il file modello dei parametri. Se non fornite questo parametro, HealthOmics crea il flusso di lavoro senza un modello di parametro.

L'esempio seguente mostra i parametri relativi al contenuto di un repository privato basato su Git:

```
aws omics create-workflow \
    --name custom-variant \
    --description "Custom variant calling pipeline" \
    --engine "WDL" \
    --definition-repository '{
            "connectionArn": "arn:aws:codeconnections:us-
east-1:123456789012:connection/abcd1234-5678-90ab-cdef-1234567890ab",
            "fullRepositoryId": "myorg/my-genomics-workflows",
            "sourceReference": {
                "type": "BRANCH",
                "value": "main"
            },
            "excludeFilePatterns": ["tests/**", "*.log"]
      }' \
    --main "workflows/variant-calling/main.wdl" \
    --parameter-template-path "parameters/variant-calling-params.json" \
    --readme-path "docs/variant-calling-README.md" \
    --storage-type "DYNAMIC" \
```

Per altri esempi, consulta il post sul blog How To Create an AWS HealthOmics Workflows from Content in Git.

Specificare un file Readme

È possibile specificare la posizione del file README utilizzando uno dei seguenti parametri:

- readme-markdown— Inserimento di una stringa o di un file sul computer locale.
- readme-uri— L'URI di un file archiviato su S3.
- readme-path — Il percorso del file README nel repository.

Utilizza readme-path solo insieme a definition-repository. Se non specificate alcun parametro README, HealthOmics importa il file README.md di livello root nel repository (se esiste).

Gli esempi seguenti mostrano come specificare la posizione del file README utilizzando readme-path e readme-uri.

```
# Using README from repository
aws omics create-workflow \
    --name "documented-workflow" \
    --definition-repository '...' \
    --readme-path "docs/workflow-guide.md"

# Using README from S3
aws omics create-workflow \
    --name "s3-readme-workflow" \
    --definition-repository '...' \
    --readme-uri "s3://my-bucket/workflow-docs/readme.md"
```

Per ulteriori informazioni, consulta HealthOmics File README del flusso di lavoro.

Specificare il file di definizione main

Se state includendo più file di definizione del flusso di lavoro, utilizzate il `main` parametro per specificare il file di definizione principale per il flusso di lavoro.

```
aws omics create-workflow  \
  --name Test  \
  --main multi_workflow/workflow2.wdl  \
    ...
```

Specificate il tipo di archiviazione di esecuzione

È possibile specificare il tipo di run storage predefinito (DYNAMIC o STATIC) e la capacità di storage di esecuzione (richiesta per lo storage statico). Per ulteriori informazioni sui tipi di run storage, consultaEsegui tipi di storage nei flussi HealthOmics di lavoro.

```
aws omics create-workflow  \
  --name my_workflow   \
  --definition-zip fileb://my-definition.zip \
  --parameter-template file://my-parameter-template.json   \
  --storage-type 'STATIC'  \
  --storage-capacity 1200  \
```

Specificare la configurazione della GPU

Utilizza il parametro accelerators per creare un flusso di lavoro che viene eseguito su un'istanza di calcolo accelerato. L'esempio seguente mostra come utilizzare il parametro. `accelerators` La configurazione della GPU viene specificata nella definizione del flusso di lavoro. Per informazioni, consulta Istanze di elaborazione accelerata.

```
aws omics create-workflow --name workflow name \
   --definition-uri s3://amzn-s3-demo-bucket1/GPUWorkflow.zip \
   --accelerators GPU
```

Configura i parametri di mappatura pull through cache

Se utilizzi la funzionalità di mappatura pull through della cache di Amazon ECR, puoi sostituire le mappature predefinite. Per ulteriori informazioni sui parametri di configurazione del contenitore, consulta. Immagini di container per flussi di lavoro privati

Nell'esempio seguente, il file `mappings.json` contiene questo contenuto:

```
{
    "registryMappings": [
        {
            "upstreamRegistryUrl": "registry-1.docker.io",
            "ecrRepositoryPrefix": "docker-hub"
        },
        {
            "upstreamRegistryUrl": "quay.io",
            "ecrRepositoryPrefix": "quay",
```

```
            "accountId": "123412341234"
        },
        {

            "upstreamRegistryUrl": "public.ecr.aws",
            "ecrRepositoryPrefix": "ecr-public"
        }
    ],
    "imageMappings": [{
            "sourceImage": "docker.io/library/ubuntu:latest",
            "destinationImage": "healthomics-docker-2/custom/ubuntu:latest",
            "accountId": "123412341234"
        },
        {
            "sourceImage": "nvcr.io/nvidia/k8s/dcgm-exporter",
            "destinationImage": "healthomics-nvidia/k8s/dcgm-exporter"
        }
    ]
}
```

Specificate i parametri di mappatura nel comando create-workflow:

```
aws omics create-workflow  \
     ...
--container-registry-map-file file://mappings.json
    ...
```

È inoltre possibile specificare la posizione S3 del file dei parametri di mappatura:

```
aws omics create-workflow  \
     ...
--container-registry-map-uri s3://amzn-s3-demo-bucket1/test.zip
    ...
```

Creazione di un flusso di lavoro utilizzando un SDK

Puoi creare un flusso di lavoro utilizzando uno dei SDKs. L'esempio seguente mostra come creare un flusso di lavoro utilizzando Python SDK.

```
import boto3

omics = boto3.client('omics')
```

```
with open('definition.zip', 'rb') as f:
   definition = f.read()

response = omics.create_workflow(
   name='my_workflow',
   definitionZip=definition,
   parameterTemplate={ ... }
)
```

Aggiornare un flusso di lavoro privato

È possibile aggiornare un flusso di lavoro utilizzando la HealthOmics console, i comandi AWS CLI o uno dei. AWS SDKs

> **Note**
>
> Non includere informazioni di identificazione personale (PII) nei nomi dei flussi di lavoro. Questi nomi sono visibili nei CloudWatch registri.

Argomenti

- Aggiornamento di un flusso di lavoro tramite la console
- Aggiornamento di un flusso di lavoro tramite la CLI
- Aggiornamento di un flusso di lavoro tramite un SDK

Aggiornamento di un flusso di lavoro tramite la console

Passaggi per aggiornare un flusso di lavoro

1. Apri la HealthOmics console.
2. Se necessario, apri il riquadro di navigazione a sinistra (≡). Scegli Flussi di lavoro privati.
3. Nella pagina Flussi di lavoro privati, scegli il flusso di lavoro da aggiornare.
4. Nella pagina Workflow:
   - Se il flusso di lavoro ha delle versioni, assicurati di selezionare la versione predefinita.
   - Scegli Modifica selezionato dall'elenco Azioni.
5. Nella pagina Modifica flusso di lavoro, puoi modificare uno qualsiasi dei seguenti valori:

- Nome del flusso di lavoro.
- Descrizione del flusso di lavoro.
- Il tipo di archiviazione Run predefinito per il flusso di lavoro.
- La capacità di archiviazione Run predefinita (se il tipo di archiviazione di esecuzione è statica). Per ulteriori informazioni sulla configurazione predefinita di Run Storage, consultaCreare un flusso di lavoro utilizzando la console.

6. Scegli Salva modifiche per applicare le modifiche.

Aggiornamento di un flusso di lavoro tramite la CLI

Come illustrato nell'esempio seguente, è possibile aggiornare il nome e la descrizione del flusso di lavoro. È inoltre possibile modificare il tipo di archiviazione di esecuzione (STATIC o DYNAMIC) e la capacità di archiviazione di esecuzione (per il tipo di archiviazione statica). Per ulteriori informazioni sui tipi di run storage, consultaEsegui tipi di storage nei flussi HealthOmics di lavoro.

```
aws omics update-workflow    \
  --id 1234567    \
  --name my_workflow      \
  --description "updated workflow"    \
  --storage-type 'STATIC'    \
  --storage-capacity 1200
```

Non ricevi una risposta alla `update-workflow` richiesta.

Aggiornamento di un flusso di lavoro tramite un SDK

Puoi aggiornare un flusso di lavoro utilizzando uno dei SDKs.

L'esempio seguente mostra come aggiornare un flusso di lavoro utilizzando Python SDK

```
import boto3

omics = boto3.client('omics')

response = omics.update_workflow(
   name='my_workflow',
   description='updated workflow'
)
```

Eliminare un flusso di lavoro privato

Quando non è più necessario un flusso di lavoro, è possibile eliminarlo utilizzando la HealthOmics console, i comandi AWS CLI o uno dei. AWS SDKs È possibile eliminare un flusso di lavoro che soddisfa i seguenti criteri:

- Il suo stato è ATTIVO o FALLITO.
- Non ha condivisioni attive.
- Hai eliminato tutte le versioni del flusso di lavoro.

L'eliminazione di un flusso di lavoro non influisce sulle esecuzioni in corso che utilizzano il flusso di lavoro.

Argomenti

Eliminazione di un flusso di lavoro tramite la console

Per eliminare un flusso di lavoro

1. Apri la HealthOmics console.
2. Se necessario, apri il riquadro di navigazione a sinistra (≡). Scegli Flussi di lavoro privati.
3. Nella pagina Flussi di lavoro privati, scegli il flusso di lavoro da eliminare.
4. Nella pagina Workflow, scegli Elimina selezionato dall'elenco Azioni.
5. Nella modalità Elimina flusso di lavoro, inserisci «conferma» per confermare l'eliminazione.
6. Scegli Elimina.

Eliminazione di un flusso di lavoro utilizzando la CLI

L'esempio seguente mostra come utilizzare il AWS CLI comando per eliminare un flusso di lavoro. Per eseguire l'esempio, sostituisci il *`workflow id`* con l'ID del flusso di lavoro che desideri eliminare.

```
aws omics delete-workflow
```

```
  --id workflow id
```

HealthOmics non invia una risposta alla `delete-workflow` richiesta.

Eliminazione di un flusso di lavoro utilizzando un SDK

Puoi eliminare un flusso di lavoro utilizzando uno dei. SDKs

L'esempio seguente mostra come eliminare un flusso di lavoro utilizzando Python SDK.

```
import boto3

omics = boto3.client('omics')

response = omics.delete_workflow(
   id='1234567'
)
```

Verifica lo stato del flusso di lavoro

Dopo aver creato il flusso di lavoro, puoi verificare lo stato e visualizzare altri dettagli del flusso di lavoro utilizzando get-workflow, come illustrato.

```
aws omics get-workflow --id 1234567
```

La risposta include i dettagli del flusso di lavoro, incluso lo stato, come mostrato.

```
{
    "arn": "arn:aws:omics:us-west-2:....",
    "creationTime": "2022-07-06T00:27:05.542459"
    "id": "1234567",
    "engine": "WDL",
    "status": "ACTIVE",
    "type": "PRIVATE",
    "main": "workflow-crambam.wdl",
    "name": "workflow_name",
    "storageType": "STATIC",
    "storageCapacity": "1200",
    "uuid": "64c9a39e-8302-cc45-0262-2ea7116d854f"
  }
```

È possibile avviare un'esecuzione utilizzando questo flusso di lavoro dopo la transizione dello stato a`ACTIVE`.

Riferimento ai file del genoma da una definizione di workflow

È possibile HealthOmics fare riferimento a un oggetto dell'archivio di riferimento con un URI come il seguente. Usa il tuo *`account ID`* *`reference store ID`* e *`reference ID`* dove indicato.

```
omics://account ID.storage.us-west-2.amazonaws.com/reference store id/reference/id
```

Alcuni flussi di lavoro richiederanno sia `SOURCE` i `INDEX` file che per il genoma di riferimento. L'URI precedente è il formato abbreviato predefinito e per impostazione predefinita sarà il file SOURCE. Per specificare uno dei due file, è possibile utilizzare il formato URI lungo, come segue.

```
omics://account ID.storage.us-west-2.amazonaws.com/reference store id/reference/id/
source
omics://account ID.storage.us-west-2.amazonaws.com/reference store id/reference/id/
index
```

L'utilizzo di un set di lettura della sequenza avrebbe uno schema simile, come mostrato.

```
aws omics create-workflow \
     --name workflow name \
     --main sample workflow.wdl \
     --definition-uri omics://account ID.storage.us-
west-2.amazonaws.com/sequence_store_id/readSet/id \
     --parameter-template file://parameters_sample_description.json
```

Alcuni set di lettura, come quelli basati su FASTQ, possono contenere letture accoppiate. Negli esempi seguenti, vengono denominati e. SOURCE1 SOURCE2 I formati come BAM e CRAM avranno solo un SOURCE1 file. Alcuni set di lettura conterranno file INDEX come i file `bai` o`crai`. L'URI precedente è il formato abbreviato predefinito e per impostazione predefinita sarà il SOURCE1 file. Per specificare il file o l'indice esatto, è possibile utilizzare il formato URI lungo, come segue.

```
omics://123456789012.storage.us-west-2.amazonaws.com/<sequence_store_id>/readSet/<id>/
source1
omics://123456789012.storage.us-west-2.amazonaws.com/<sequence_store_id>/readSet/<id>/
source2
```

```
omics://123456789012.storage.us-west-2.amazonaws.com/<sequence_store_id>/readSet/<id>/
index
```

Di seguito è riportato un esempio di file JSON di input che utilizza due Omics Storage. URIs

```
{
   "input_fasta": "omics://123456789012.storage.us-west-2.amazonaws.com/
<reference_store_id>/reference/<id>",
   "input_cram": "omics://123456789012.storage.us-west-2.amazonaws.com/
<sequence_store_id>/readSet/<id>"
}
```

Fai riferimento al file JSON di input in AWS CLI aggiungendolo **`--inputs file://<input_file.json>`** alla tua richiesta start-run.

Controllo delle versioni del flusso di lavoro in HealthOmics

Se devi apportare modifiche a un flusso di lavoro, puoi creare un nuovo flusso di lavoro o una nuova versione del flusso di lavoro. Le versioni sono immutabili, ad eccezione delle modifiche di configurazione consentite che non influiscono sulla logica di esecuzione.

Le versioni Workflow offrono i seguenti vantaggi:

- Le versioni costituiscono un gruppo logico di flussi di lavoro correlati. È possibile aggiungere un nome definito dall'utente a ciascuna versione del flusso di lavoro per gestirla più facilmente (soprattutto per un flusso di lavoro con un numero elevato di versioni).
- È possibile eseguire più versioni di un flusso di lavoro contemporaneamente.
- Tutte le versioni di un flusso di lavoro condividono lo stesso ID del flusso di lavoro e lo stesso ARN di base, il che può semplificare la gestione della pipeline dopo la modifica di un flusso di lavoro.
- Le versioni del flusso di lavoro forniscono lo stesso livello di provenienza dei dati dei flussi di lavoro. Le versioni sono immutabili e HealthOmics creano un ARN univoco per ogni versione del flusso di lavoro. L'ARN della versione include l'ID del flusso di lavoro e il nome della versione, come illustrato nell'esempio seguente:

  ```
  arn:aws:omics:us-west-2:123456789012:workflow/1234567/version/
  myUniqueVersionName
  ```
- Se possiedi un flusso di lavoro condiviso, puoi aggiornare il flusso di lavoro senza interrompere gli abbonati (che possono continuare a utilizzare la versione precedente). Gli abbonati possono

accedere a tutte le versioni del flusso di lavoro. Se crei una nuova versione, non è necessario condividere nuovamente il flusso di lavoro.

- Quando si avvia l'esecuzione di un flusso di lavoro, è possibile specificare la versione del flusso di lavoro.
  - Gli utenti possono scegliere di mantenere una versione stabile per le esecuzioni di produzione e provare la versione più recente per un'esecuzione di prova.
  - Gli utenti possono tornare alla versione precedente di un flusso di lavoro se riscontrano problemi con la nuova versione.
  - Gli abbonati a un flusso di lavoro condiviso possono scegliere quale versione utilizzare.

Argomenti

Versione predefinita del flusso di lavoro

Dopo aver creato una o più versioni di un flusso di lavoro, HealthOmics considera il flusso di lavoro originale come versione predefinita. Quando si avvia un'esecuzione, è possibile facoltativamente specificare una versione del flusso di lavoro per l'esecuzione. Se non si specifica una versione all'avvio di un'esecuzione, HealthOmics utilizza la versione predefinita.

Nella console, HealthOmics indica il flusso di lavoro originale con un'etichetta di versione predefinita. La console utilizza questa etichetta solo dopo aver creato una o più versioni del flusso di lavoro. Il flusso di lavoro originale rimane sempre la versione predefinita. Non puoi assegnare nessun'altra versione come predefinita.

Non è possibile eliminare la versione predefinita di un flusso di lavoro se vi sono altre versioni associate al flusso di lavoro. Per ulteriori informazioni, consulta Eliminare un flusso di lavoro privato.

Creare una versione del flusso di lavoro

Quando crei una nuova versione di un flusso di lavoro, devi specificare i valori di configurazione per la nuova versione. Non eredita alcun valore di configurazione dal flusso di lavoro.

Quando crei la versione, fornisci un nome di versione univoco per questo flusso di lavoro. Non è possibile modificare il nome dopo aver HealthOmics creato la versione.

Il nome della versione deve iniziare con una lettera o un numero e può includere lettere maiuscole e minuscole, numeri, trattini, punti e caratteri di sottolineatura. La lunghezza massima è di 64 caratteri. Ad esempio, è possibile utilizzare uno schema di denominazione semplice, ad esempio version1, version2, version3. È inoltre possibile abbinare le versioni del flusso di lavoro alle proprie convenzioni interne di controllo delle versioni, ad esempio 2.7.0, 2.7.1, 2.7.2.

Facoltativamente, utilizzate il campo di descrizione della versione per aggiungere note su questa versione. Ad esempio: Fix for syntax error in workflow definition.

> **Note**
>
> Non includere informazioni di identificazione personale (PII) nel nome della versione. I nomi delle versioni vengono visualizzati nella versione del workflow ARN.

HealthOmics assegna un ARN univoco alla versione del flusso di lavoro. L'ARN è unico in base alla combinazione di ID del flusso di lavoro e nome della versione.

> **Warning**
>
> Dopo aver eliminato una versione del flusso di lavoro, HealthOmics consente di riutilizzare il nome della versione per una versione diversa del flusso di lavoro. È consigliabile non riutilizzare i nomi delle versioni. Se riutilizzi un nome, il flusso di lavoro e ogni versione hanno un UUID univoco che puoi usare come provenienza.

Argomenti

Crea una versione del flusso di lavoro utilizzando la console

Passaggi per creare una versione del flusso di lavoro

1. Apri la HealthOmics console.
2. Se necessario, apri il riquadro di navigazione a sinistra (≡). Scegli Flussi di lavoro privati.
3. Nella pagina Flussi di lavoro privati, scegli il flusso di lavoro per la nuova versione.
4. Nella pagina dei dettagli del flusso di lavoro, scegli Crea nuova versione.
5. Nella pagina Crea versione, fornisci le seguenti informazioni:
    1. Nome della versione: inserisci un nome per la versione del flusso di lavoro che sia univoco in tutto il flusso di lavoro.
    2. Descrizione della versione (opzionale): puoi utilizzare il campo della descrizione per aggiungere note su questa versione.
6. Nel pannello di definizione del flusso di lavoro, fornite le seguenti informazioni:
    1. Lingua del flusso di lavoro (opzionale): seleziona la lingua delle specifiche per la versione del flusso di lavoro. Altrimenti, HealthOmics determina la lingua dalla definizione del flusso di lavoro.
    2. Per l'origine delle definizioni di Workflow, scegli di importare la cartella delle definizioni da un repository basato su Git, una posizione Amazon S3 o da un'unità locale.
        a. Per l'importazione da un servizio di repository:

> **Note**
>
> HealthOmics supporta repository pubblici e privati perGitHub,,, GitLabBitbucket,GitHub self-managed. GitLab self-managed

        i. Scegli una connessione per connettere AWS le tue risorse al repository esterno. Per creare una connessione, consultaConnect con repository di codice esterni.

> **Note**
>
> I clienti della TLV regione devono creare una connessione nella regione IAD (us-east-1) per creare un flusso di lavoro.

ii. In Full repository ID, inserisci l'ID del repository come nome utente/nome del repository. Verifica di avere accesso ai file in questo repository.

iii. In Riferimento alla fonte (opzionale), inserisci un riferimento alla fonte del repository (branch, tag o ID di commit). HealthOmics utilizza il ramo predefinito se non viene specificato alcun riferimento alla fonte.

iv. In Escludi modelli di file, inserisci i modelli di file per escludere cartelle, file o estensioni specifici. Questo aiuta a gestire le dimensioni dei dati durante l'importazione dei file del repository. Esistono un massimo di 50 pattern e i pattern devono seguire la sintassi del modello glob. Ad esempio:

A. `tests/`

B. `*.jpeg`

C. `large_data.zip`

b. Per Seleziona la cartella di definizione da S3:

i. Inserisci la posizione Amazon S3 che contiene la cartella di definizione del flusso di lavoro compressa. Il bucket Amazon S3 deve trovarsi nella stessa regione del flusso di lavoro.

ii. Se il tuo account non possiede il bucket Amazon S3, inserisci l'ID dell'account del proprietario del bucket nell'ID dell' AWS account del proprietario del bucket S3. Queste informazioni sono necessarie per verificare la proprietà del HealthOmics bucket.

c. Per Seleziona la cartella di definizione da una fonte locale:

i. Immettete la posizione sull'unità locale della cartella di definizione del flusso di lavoro compressa.

3. Percorso principale del file di definizione del flusso di lavoro (opzionale): immettete il percorso del file dalla cartella o dall'archivio di definizione del flusso di lavoro compresso al file. `main` Questo parametro non è richiesto se è presente un solo file nella cartella di definizione del flusso di lavoro o se il file principale è denominato «principale».

7. Nel pannello del file README (opzionale), selezionate l'origine del file README e fornite le seguenti informazioni:

- Per Importazione da un servizio di repository, nel percorso del file README, immettete il percorso del file README all'interno del repository.
- Per Seleziona file da S3, nel file README in S3, inserisci l'URI Amazon S3 per il file README.

- Per Seleziona file da una fonte locale: in README - opzionale, scegli Scegli file per selezionare il file markdown (.md) da caricare.

8. Nel pannello di configurazione dell'archiviazione di esecuzione predefinita, fornisci il tipo e la capacità di archiviazione di esecuzione predefiniti per le esecuzioni che utilizzano questo flusso di lavoro:

    1. Tipo di archiviazione di esecuzione: scegli se utilizzare l'archiviazione statica o dinamica come impostazione predefinita per l'archiviazione di esecuzione temporanea. L'impostazione predefinita è l'archiviazione statica.
    2. Capacità di archiviazione di esecuzione (opzionale): per il tipo di storage di esecuzione statico, è possibile inserire la quantità predefinita di storage di esecuzione richiesta per questo flusso di lavoro. Il valore predefinito per questo parametro è 1200 GiB. È possibile sovrascrivere questi valori predefiniti quando si avvia un'esecuzione.

9. Tag (opzionale): puoi associare fino a 50 tag a questa versione del flusso di lavoro.
10. Scegli Next (Successivo).
11. Nella pagina Aggiungi parametri del flusso di lavoro (opzionale), seleziona l'origine dei parametri:

    1. Per Analizza dal file di definizione del flusso di lavoro, HealthOmics analizzerà automaticamente i parametri del flusso di lavoro dal file di definizione del flusso di lavoro.
    2. Per Fornire un modello di parametro dal repository Git, usa il percorso del file del modello di parametro dal tuo repository.
    3. Per Seleziona il file JSON dalla fonte locale, carica un JSON file da una fonte locale che specifichi i parametri.
    4. In Inserisci manualmente i parametri del flusso di lavoro, inserisci manualmente i nomi e le descrizioni dei parametri.

12. Nel pannello di anteprima dei parametri, è possibile rivedere o modificare i parametri per questa versione del flusso di lavoro. Se ripristinate il JSON file, perderete tutte le modifiche locali apportate.
13. Nella pagina di rimappatura dell'URI del contenitore, nel pannello Regole di mappatura, puoi definire le regole di mappatura URI per il tuo flusso di lavoro.

    Per Sorgente del file di mappatura, selezionate una delle seguenti opzioni:

    - Nessuna: non è richiesta alcuna regola di mappatura.
    - Seleziona il file JSON da S3: specifica la posizione S3 per il file di mappatura.

- Seleziona il file JSON da una fonte locale: specifica la posizione del file di mappatura sul tuo dispositivo locale.
- Inserisci manualmente le mappature: inserisci le mappature del registro e le mappature delle immagini nel pannello Mappature.

14. La console visualizza il pannello Mappature. Se avete scelto un file sorgente di mappatura, la console visualizza i valori del file.

    a. Nelle mappature del registro, è possibile modificare le mappature o aggiungere mappature (massimo 20 mappature del registro).

       Ogni mappatura del registro contiene i seguenti campi:

       - URL del registro upstream: l'URI del registro upstream.
       - Prefisso del repository ECR: il prefisso del repository da utilizzare nell'archivio privato Amazon ECR.
       - (Facoltativo) Prefisso del repository upstream: il prefisso del repository nel registro upstream.
       - (Facoltativo) ID account ECR: ID account dell'account proprietario dell'immagine del contenitore upstream.

    b. Nelle mappature delle immagini, è possibile modificare le mappature delle immagini o aggiungere mappature (massimo 100 mappature di immagini).

       Ogni mappatura di immagini contiene i seguenti campi:

       - Immagine di origine: specifica l'URI dell'immagine di origine nel registro upstream.
       - Immagine di destinazione: specifica l'URI dell'immagine corrispondente nel registro privato di Amazon ECR.

15. Scegli Next (Successivo).
16. Controlla la configurazione della versione, quindi scegli Crea versione.

Una volta creata la versione, la console torna alla pagina dei dettagli del flusso di lavoro e visualizza la nuova versione nella tabella Flussi di lavoro e versioni.

Creare una versione del flusso di lavoro utilizzando la CLI

È possibile creare una versione del flusso di lavoro utilizzando l'operazione `CreateWorkflowVersion` API. Per i parametri opzionali, HealthOmics utilizza le seguenti impostazioni predefinite:

Parametro	Default
Motore	Determinato dalla definizione del flusso di lavoro
Storage Type (Tipo di storage)	STATIC
Capacità di archiviazione (per l'archiviazione statica)	1200 GiB
Principale	Determinato in base al contenuto della cartella di definizione del flusso di lavoro. Per informazioni dettagliate, vedi HealthOmics requisiti di definizione del flusso di lavoro.
Acceleratori	nessuno
Tag	nessuno

L'esempio CLI seguente crea una versione del flusso di lavoro con archiviazione statica come archiviazione di esecuzione predefinita:

```
aws omics create-workflow-version \
--workflow-id 1234567  \
--version-name "my_version" \
--engine WDL \
--definition-zip fileb://workflow-crambam.zip \
--description "my version description" \
--main file://workflow-params.json \
--parameter-template file://workflow-params.json \
--storage-type='STATIC'   \
--storage-capacity 1200   \
--tags example123=string  \
```

```
--accelerators GPU
```

Se il file di definizione del flusso di lavoro si trova in una cartella Amazon S3, inserisci la posizione utilizzando il `definition-uri` parametro anziché. `definition-zip` Per ulteriori informazioni, CreateWorkflowVersionconsulta AWS HealthOmics API Reference.

Riceverai la seguente risposta alla `create-workflow-version` richiesta.

```
{
  "workflowId": "1234567",
  "versionName": "my_version",
  "arn": "arn:aws:omics:us-west-2:123456789012:workflow/1234567/version/3",
  "status": "ACTIVE",
  "tags": {
    "environment": "production",
    "owner": "team-alpha"
  },
  "uuid": "0ac9a563-355c-fc7a-1b47-a115167af8a2"
}
```

Crea una versione del flusso di lavoro utilizzando un SDK

Puoi creare un flusso di lavoro utilizzando uno dei SDKs.

L'esempio seguente mostra come creare una versione del flusso di lavoro utilizzando Python SDK.

```
import boto3

omics = boto3.client('omics')

with open('definition.zip', 'rb') as f:
   definition = f.read()

response = omics.create_workflow_version(
   workflowId='1234567',
   versionName='my_version',
   requestId='my_request_1'
   definitionZip=definition,
   parameterTemplate={ ... }
)
```

Verifica lo stato di una versione del flusso di lavoro

Dopo aver creato la versione del flusso di lavoro, è possibile verificare lo stato e visualizzare altri dettagli del flusso di lavoro utilizzando get-workflow-version, come illustrato.

```
aws omics get-workflow-version
--workflow-id 9876543
--version-name "my_version"
```

La risposta fornisce i dettagli del flusso di lavoro, incluso lo stato, come mostrato.

```
{
  "workflowId": "1234567",
  "versionName": "3.0.0",
  "arn": "arn:aws:omics:us-west-2:123456789012:workflow/1234567/version/3.0.0",
  "status": "ACTIVE",
  "description": ...
  "uuid": "0ac9a563-355c-fc7a-1b47-a115167af8a2"
}
```

Prima di poter iniziare un'esecuzione con questa versione del flusso di lavoro, lo stato deve passare a`ACTIVE`.

Aggiornare una versione del flusso di lavoro

Puoi aggiornare la descrizione e la configurazione di archiviazione di esecuzione predefinita per una versione privata del flusso di lavoro. Per modificare qualsiasi altra informazione nella versione del flusso di lavoro, crea una nuova versione.

Argomenti

- Aggiorna una versione del flusso di lavoro utilizzando la console
- Aggiornare una versione del flusso di lavoro utilizzando la CLI
- Aggiorna una versione del flusso di lavoro utilizzando un SDK

Aggiorna una versione del flusso di lavoro utilizzando la console

Per aggiornare una versione del flusso di lavoro

1. Apri la HealthOmics console.

2. Se necessario, apri il riquadro di navigazione a sinistra (≡). Scegli Flussi di lavoro privati.
3. Nella pagina Flussi di lavoro privati, scegli il flusso di lavoro.
4. Nella pagina Flusso di lavoro, scegli la versione del flusso di lavoro da aggiornare e scegli Modifica, selezionato dall'elenco Azioni.

   - Se scegli la versione predefinita, la console apre la pagina Modifica flusso di lavoro. Per ulteriori informazioni, consulta Aggiornare un flusso di lavoro privato.
   - Se si sceglie una versione definita dall'utente, la console apre la pagina Modifica versione.
5. Nella pagina Modifica versione, fornisci le seguenti informazioni

   - Descrizione della versione (opzionale): una descrizione di questa versione.
6. Nel pannello Default Run Storage Configuration, fornite i seguenti valori predefiniti per le esecuzioni che utilizzano questa versione del flusso di lavoro. È possibile sovrascrivere i valori predefiniti quando si avvia un'esecuzione:

   - Per il tipo di archiviazione Run, seleziona Statico o Dinamico.
   - Per l'archiviazione statica in esecuzione, seleziona la quantità predefinita di capacità di archiviazione Run per le esecuzioni che utilizzano questa versione del flusso di lavoro. Il valore predefinito per questo parametro è 1200 GiB.
7. Scegli Save changes (Salva modifiche).

La console torna alla pagina dei dettagli del flusso di lavoro e visualizza un banner di pagina con la versione aggiornata del flusso di lavoro.

Aggiornare una versione del flusso di lavoro utilizzando la CLI

È possibile aggiornare i parametri per una versione del flusso di lavoro utilizzando il seguente comando CLI. La combinazione di ID del workflow e nome della versione identifica in modo univoco la versione.

```
aws omics update-workflow-version
--workflow-id 1234567
--version-name "my_version"
--storage-type 'STATIC'
--storage-capacity 2400
--description "version description"
```

Non riceverai alcuna risposta alla `update-workflow-version` richiesta.

Aggiorna una versione del flusso di lavoro utilizzando un SDK

Puoi aggiornare una versione del flusso di lavoro utilizzando uno dei SDKs. Il seguente esempio di Python SDK mostra come aggiornare il tipo di archiviazione e la descrizione per una versione del flusso di lavoro.

```
import boto3

omics = boto3.client('omics')

response = omics.update_workflow_version(
   workflowID=1234567,
   versionName='3.0.0',
   storageType='DYNAMIC',
   description='new version description'
)
```

Eliminare una versione del flusso di lavoro

È possibile eliminare una versione del flusso di lavoro definita dall'utente utilizzando la console, la CLI o uno dei. SDKs L'eliminazione di una versione del flusso di lavoro non influisce sulle esecuzioni in corso che utilizzano la versione del flusso di lavoro.

Non è possibile eliminare il. Versione predefinita del flusso di lavoro Elimini tutte le versioni definite dall'utente, quindi elimini il flusso di lavoro.

Argomenti

- Eliminare una versione del flusso di lavoro utilizzando la console
- Eliminare una versione del flusso di lavoro utilizzando la CLI
- Eliminare una versione del flusso di lavoro utilizzando un SDK

Eliminare una versione del flusso di lavoro utilizzando la console

Per eliminare una versione del flusso di lavoro

1. Apri la HealthOmics console.
2. Se necessario, apri il riquadro di navigazione a sinistra (≡). Scegli Flussi di lavoro privati.
3. Nella pagina Flussi di lavoro privati, scegli il flusso di lavoro.

4. Nella pagina Flusso di lavoro, scegli la versione del flusso di lavoro da eliminare e scegli Elimina selezionata dall'elenco Azioni.
5. Nella finestra di dialogo Elimina la versione del flusso di lavoro, inserisci «conferma» per confermare l'eliminazione.
6. Scegli Elimina.

La console visualizza un banner di pagina con la versione del flusso di lavoro eliminata.

Eliminare una versione del flusso di lavoro utilizzando la CLI

È possibile eliminare una versione del flusso di lavoro definita dall'utente utilizzando il seguente comando CLI. La combinazione di ID del workflow e nome della versione identifica in modo univoco la versione.

```
aws omics delete-workflow-version
--workflow-id 9876543
--version-name "my_version"
```

Non riceverai alcuna risposta alla `delete-workflow-version` richiesta.

Eliminare una versione del flusso di lavoro utilizzando un SDK

Puoi eliminare un flusso di lavoro utilizzando uno dei SDKs.

L'esempio seguente mostra come eliminare un flusso di lavoro utilizzando Python SDK.

```
import boto3

omics = boto3.client('omics')

response = omics.delete_workflow_version(
   workflowID=1234567,
   versionName='3.0.0'
)
```

Utilizzo delle HealthOmics corse

Dopo aver creato un flusso di lavoro, puoi avviare le esecuzioni utilizzando il flusso di lavoro.

Quando si avvia un'esecuzione, HealthOmics alloca lo spazio di archiviazione temporaneo di esecuzione per il motore del flusso di lavoro da utilizzare durante l'esecuzione. Per garantire l'isolamento e la sicurezza dei dati, effettua il HealthOmics provisioning dello storage all'inizio di ogni esecuzione e lo disattiva alla fine dell'esecuzione.

HealthOmics fornisce diverse quote relative alle esecuzioni e alle attività del flusso di lavoro. I valori predefiniti sono intenzionalmente conservativi, per evitare sforamenti imprevisti dei costi. È possibile richiedere un aumento di queste quote. Per ulteriori informazioni, consulta HealthOmics quote di servizio.

Quando avvii un'esecuzione, HealthOmics assegna un run ID e un run uuid alla corsa. Le esecuzioni in un account hanno un'esecuzione unica. IDs Tuttavia, HealthOmics riutilizza l'esecuzione eliminata IDs, quindi una corsa e una corsa eliminata possono avere lo stesso ID di esecuzione. Inoltre, è raro ma possibile che un flusso di lavoro condiviso abbia lo stesso ID di esecuzione di una esecuzione nel tuo account.

run uuidÈ un identificatore univoco globale (GUID) che puoi utilizzare per identificare le esecuzioni tra account o per distinguere tra due esecuzioni nel tuo account che hanno lo stesso ID di esecuzione.

> **Note**
>
> Ai fini della provenienza dei dati, ti consigliamo di utilizzare il comando per identificare in modo univoco run uuid le esecuzioni. run uuidÈ anche l'identificatore migliore da collegare al sistema interno di gestione delle informazioni di laboratorio (LIMs) o al sistema di tracciamento dei campioni.

Puoi utilizzare Amazon Q CLI per ottimizzare le tue esecuzioni e analizzare le prestazioni di esecuzione. Per ulteriori informazioni, consulta le istruzioni di esempio per la CLI di Amazon Q e il tutorial sull'intelligenza artificiale generativa HealthOmics Agentic su. GitHub

Argomenti

- Esegui tipi di storage nei flussi HealthOmics di lavoro
- Esegui la modalità di conservazione per le HealthOmics esecuzioni
- HealthOmics eseguire ingressi
- Esegui il ciclo di vita in un flusso di lavoro HealthOmics
- HealthOmics esegui uscite
- Motivi dell'errore di esecuzione

- Ciclo di vita delle attività in esecuzione HealthOmics
- Esegui l'ottimizzazione per un HealthOmics flusso di lavoro privato
- Esegui operazioni in HealthOmics

Esegui tipi di storage nei flussi HealthOmics di lavoro

Quando avvii un'esecuzione, HealthOmics alloca lo storage di esecuzione temporaneo per il motore di workflow da utilizzare durante l'esecuzione. HealthOmicsfornisce l'archiviazione temporanea di esecuzione come file system.

Per un determinato flusso di lavoro o esecuzione del flusso di lavoro, è possibile scegliere l'archiviazione di esecuzione dinamica o statica. Per impostazione predefinita, HealthOmics fornisce un'archiviazione di esecuzione DINAMICA.

> **Note**
>
> L'utilizzo di Run Storage comporta addebiti sul tuo account. Per informazioni sui prezzi dello storage a esecuzione statica e dinamica, consulta la pagina HealthOmicsdei prezzi.

Le sezioni seguenti forniscono informazioni da considerare quando si decide quale tipo di run storage utilizzare.

Run Storage dinamico

Si consiglia di utilizzare lo storage a esecuzione dinamica per la maggior parte delle esecuzioni, comprese quelle che richiedono tempi di avvio più rapidi, quelle in cui non si conoscono in anticipo le esigenze di storage e per cicli iterativi di test di sviluppo.

Non è necessario stimare lo storage o la velocità effettiva richiesti per l'esecuzione. HealthOmics aumenta o riduce dinamicamente le dimensioni dello storage, in base all'utilizzo del file system durante l'esecuzione. HealthOmics inoltre ridimensiona dinamicamente la velocità effettiva in base alle esigenze del flusso di lavoro. Un'esecuzione non fallisce mai a causa di un errore di archiviazione insufficiente per il file system.

Lo storage a esecuzione dinamica offre provisioning/deprovisioning tempi più rapidi rispetto allo storage a esecuzione statica. Una configurazione più rapida è un vantaggio per la maggior parte dei flussi di lavoro e lo è anche durante development/test i cicli.

Al termine dell'esecuzione (percorso di successo o percorso di errore), l'operazione API GetRun restituisce lo spazio di archiviazione massimo utilizzato dall'esecuzione nel campo StorageCapacity. È inoltre possibile trovare queste informazioni nei log del manifesto di esecuzione che si trovano nel gruppo di log. omics Per un'esecuzione dinamica dello storage che viene completata entro 2 ore, il valore massimo di archiviazione potrebbe non essere disponibile.

Per lo storage a esecuzione dinamica, il run esegue il provisioning di un file system che utilizza il protocollo NFS. NFS considera le operazioni CREATE, DELETE e RENAME sui file come operazioni non idempotenti, il che può occasionalmente portare a condizioni di concorrenza per queste operazioni che il codice deve gestire correttamente. Ad esempio, il codice non dovrebbe fallire se tenta di eliminare un file che non esiste. Prima di adottare lo storage a esecuzione dinamica, consigliamo di modificare il codice del flusso di lavoro per renderlo resiliente a operazioni sui file non idempotenti. Per informazioni, consulta Esempi di codice per la gestione sicura di operazioni non idempotenti.

Esempi di codice per la gestione sicura di operazioni non idempotenti

Il seguente esempio di python mostra come eliminare un file senza errori se il file non esiste.

```
import os
import errno

def remove_file(file_path):
    try:
        os.remove(file_path)
    except OSError as e:
        # If the error is "No such file or directory", ignore it (or log it)
        if e.errno != errno.ENOENT:
            # Otherwise, raise the error
            raise

# Example usage
remove_file("myfile")
```

I seguenti esempi utilizzano la shell Bash. Per rimuovere in modo sicuro un file anche se non esiste, usa:

```
rm -f my_file
```

Per spostare (rinominare) un file in modo sicuro, esegui il comando move solo se il file `old_name` esiste nella directory corrente.

```
[ -f old_name ] && mv old_name new_name
```

Per creare una directory, utilizzate il seguente comando:

```
mkdir -p mydir/subdir/
```

Archiviazione statica di esecuzione

Per l'archiviazione statica in esecuzione, run fornisce un file system che utilizza il protocollo Lustre. Per impostazione predefinita, questo protocollo è resiliente alle operazioni sui file non idempotenti. Non è necessario modificare il codice del flusso di lavoro per gestire operazioni sui file non idempotenti.

HealthOmics alloca una quantità fissa di storage di esecuzione. Questo valore viene specificato all'avvio della corsa. Lo storage di esecuzione predefinito è 1200 GiB, se non si specifica un valore. Quando si specifica un valore per la dimensione di archiviazione nella richiesta StartRun API, il sistema arrotonda il valore al multiplo più vicino di 1200 GiB. Se tale dimensione di archiviazione non è disponibile, viene arrotondata al multiplo più vicino di 2400 GiB.

Per lo storage a esecuzione statica, fornisce i seguenti HealthOmics valori di throughput:

- Throughput di base di 200 per MB/s TiB di capacità di storage fornita.
- Throughput burst fino a 1300 per MB/s TiB di capacità di storage fornita.

Se la dimensione di storage specificata è troppo bassa, l'esecuzione ha esito negativo e viene visualizzato un errore Out of storage for file system. Lo storage a esecuzione statica è ideale per flussi di lavoro prevedibili con requisiti di storage noti.

Lo storage a esecuzione statica è adatto per carichi di lavoro di grandi dimensioni e con elevata contemporaneità delle attività (ad esempio, un grande volume di campioni RNASeq elaborati in parallelo). Fornisce un throughput di file system per GiB più elevato e un costo per GiB inferiore rispetto allo storage a esecuzione dinamica.

Calcolo dello storage di esecuzione statico richiesto

Un flusso di lavoro richiede una capacità aggiuntiva quando utilizza lo storage di esecuzione statico (rispetto allo storage a esecuzione dinamica) perché l'installazione del file system di base utilizza il 7% della capacità statica del file system.

Se esegui un workflow di esecuzione dinamica per misurare lo storage massimo utilizzato dall'esecuzione, utilizza il seguente calcolo per determinare la quantità minima di storage statico richiesta:

```
static storage required =
       maximum storage in GiB used by the dynamic run storage
         + (total static file system size in GiB * 0.07)
```

Esempio:

```
    Maximum storage measured from a dynamic run storage workflow run:  500GiB
      File system size: 1200GiB
      7% of the file system size:  84GiB
      500 + 84 = 584GiB of static run storage required for this run.
```

Pertanto, 1200 GiB (la capacità minima per lo storage di esecuzione statica) sono sufficienti per questa esecuzione.

Esegui la modalità di conservazione per le HealthOmics esecuzioni

Al termine di un'esecuzione, HealthOmics archivia i metadati di esecuzione in. CloudWatch Per impostazione predefinita, CloudWatch conserva i dati di esecuzione a tempo indeterminato, a meno che non si modifichi la CloudWatch politica di conservazione. Gli output di esecuzione vengono inoltre archiviati in Amazon S3 fino a quando non vengono eliminati.

Una delle opzioni regolabili HealthOmics quote di servizio è quella maximum number of runs (active and inactive) in una regione. HealthOmics conserva i metadati di esecuzione per un massimo di questo numero di esecuzioni per l'utilizzo da parte della console e delle operazioni API (ListRuns e). GetRun Quando si avvia un'esecuzione, è possibile impostare il parametro della modalità di conservazione dell'esecuzione per indicare il comportamento di conservazione dell'esecuzione. Il parametro supporta i valori REMOVE e RETAIN.

Per una nuova esecuzione con la modalità di conservazione impostata su REMOVE, se HealthOmics tenta di aggiungere la corsa dopo aver già salvato il numero massimo di esecuzioni, rimuove

automaticamente i metadati per l'esecuzione più vecchia che ha impostato la modalità REMOVE. Questa rimozione non influisce sui dati archiviati in Amazon S3 CloudWatch o Amazon S3.

RETAIN è il valore predefinito per la modalità run retention. Per le esecuzioni in questa modalità, il sistema non elimina i metadati di esecuzione. Se si HealthOmics raggiunge il numero massimo di esecuzioni, tutte impostate su RETAIN, non sarà possibile creare esecuzioni aggiuntive finché non si eliminano alcune esecuzioni.

Se hai intenzione di eseguire un batch di più del numero massimo di esecuzioni contemporaneamente, assicurati di impostare la modalità di mantenimento dell'esecuzione su REMOVE. In caso contrario, il batch ha esito negativo quando HealthOmics tenta di avviare l'esecuzione successiva dopo il massimo.

Considerazioni aggiuntive sull'utilizzo della modalità di conservazione REMOVE:

- Quando inizi a utilizzare REMOVE come modalità di conservazione, valuta la possibilità di eliminare una o più esecuzioni che utilizzano la modalità RETAIN, per liberare slot. Quando si avviano altre esecuzioni REMOVE, subentra la rimozione automatica, quindi sono disponibili abbastanza slot per nuove esecuzioni.
- Se desideri eseguire nuovamente un'esecuzione archiviata (o un insieme di esecuzioni), utilizza lo strumento HealthOmics Rerun CLI. Per ulteriori informazioni ed esempi su come utilizzare questo strumento, consulta Omics rerun nel repository degli strumenti. HealthOmics GitHub
- Ti consigliamo di configurare un nome univoco per ogni esecuzione. Dopo aver HealthOmics rimosso un'esecuzione, non puoi utilizzare la console o l'API per trovare il nome o l'ID di esecuzione. Tuttavia, puoi utilizzarlo per CloudWatch cercare il nome della corsa, quindi usa nomi univoci per ottenere i migliori risultati di ricerca.
- È possibile utilizzare il CloudWatch start-query comando per ottenere informazioni su un'esecuzione archiviata. Se il nome dell'esecuzione non è univoco, la query può restituire più manifesti. I parametri di inizio e fine definiscono l'intervallo di tempo per la ricerca.

```
aws logs start-query  \
    --log-group-name "/aws/omics/WorkflowLog" \
    --query-string 'filter @logStream like "manifest" and @message like "myRunName"'
 \
    --end-time <END-EPOCH-TIME> --start-time <START-EPOCH-TIME>
```

  Il start-query comando restituisce un ID di interrogazione. Il passaggio dell'ID della query al get-query-results comando restituisce i risultati della query.

```
aws logs get-query-results --query-id QueryId
```

HealthOmics eseguire ingressi

Se la definizione del flusso di lavoro specifica i file di input per il flusso di lavoro o le attività del flusso di lavoro HealthOmics , inserisce i file in un volume di memoria virtuale dedicato all'esecuzione del flusso di lavoro. Questi file di input sono di sola lettura, il che impedisce alle attività di modificare i potenziali input in altre attività del flusso di lavoro. Per le importazioni di directory, le directory sono anche di sola lettura.

Molte applicazioni di genomica presuppongono che i file indice siano collocati insieme ai file di sequenza (ad esempio un file associato `bai` a un file). `bam` Per includere i file indice, specificateli come input delle attività nella definizione del flusso di lavoro.

Argomenti

Gestione delle dimensioni dei parametri di esecuzione

Quando si avvia un'esecuzione, si specificano gli input di esecuzione nell'oggetto o file JSON dei parametri di esecuzione. È possibile specificare fino a 50 KB di parametri di esecuzione per il flusso di lavoro. È possibile utilizzare le seguenti tecniche per rimanere entro questo limite di dimensione:

- Utilizzate le importazioni di directory

  Per specificare un numero elevato di file di input, specifica un parametro come posizione Amazon S3 che contiene tutti i file, anziché specificare un parametro per ogni posizione di file. Per ulteriori informazioni, consulta l'argomento successivo (Formati dei parametri di input di Amazon S3).
- Usa un foglio di esempio

  Un foglio di esempio è un file CSV o TSV con una colonna per l'indirizzo fastq.gz (o due per la lettura abbinata) e colonne aggiuntive per i metadati, ad esempio i nomi di esempio. Il foglio di esempio viene specificato come parametro di input di esecuzione anziché come parametro per ogni file di input.

Il flusso di lavoro definisce il modo in cui il foglio di esempio viene mappato alle strutture di dati del flusso di lavoro. Sebbene sia possibile scrivere codice per fogli di esempio in WDL e CWL, questi sono più comuni in. NextFlow Per un esempio, consultate il foglio di esempio sul sito GitHub nf-core.

Formati dei parametri di input di Amazon S3

Per un parametro di input che accetta una posizione Amazon S3, il parametro può specificare la posizione di un file o di un'intera directory di file. L'utilizzo di una directory presenta i seguenti vantaggi:

- Comodità: si specifica il nome della directory come parametro. Non si elencano tutti i nomi di file.
- Compattezza: la dimensione massima del file del parametro di input è 50 KB. Se si fornisce un lungo elenco di nomi di file di input, è possibile superare questo limite.

Amazon S3 è un sistema di storage di oggetti piatto, quindi non supporta le directory. I file vengono raggruppati in una «directory» assegnando a ciascun file lo stesso prefisso key dell'oggetto. Per ulteriori informazioni sui prefissi delle chiavi degli oggetti di Amazon S3, consulta Organizzazione degli oggetti utilizzando i prefissi.

HealthOmics interpreta il valore del parametro di input come segue:

- Se la posizione di Amazon S3 non termina con una barra o utilizza il modello a glob, HealthOmics prevede che il valore del parametro sia la chiave per un oggetto Amazon S3.

  Ad esempio, si specifica di inserire file1.fastq `s3://myfiles/runs/inputs/a/file1.fastq`
- Se la posizione Amazon S3 termina con una barra, HealthOmics interpreta il valore del parametro come un prefisso Amazon S3. Carica tutti gli oggetti Amazon S3 con quel prefisso.

  Ad esempio, puoi specificare `s3://myfiles/runs/inputs/a/` di caricare tutti gli oggetti le cui chiavi iniziano con questo prefisso.
- Per Nextflow, supporta HealthOmics parzialmente il modello glob per Amazon S3 nei parametri di input. URIs

  Ad esempio, puoi specificare di inserire tutti i file.gz le `“s3://myfiles/runs/inputs/a/*.gz”` cui chiavi iniziano con questo prefisso.

Gestione Nextflow del pattern Glob negli input di Amazon S3

Modello Glob	HealthOmics Comportamento della partita	Note
s3://bucket/directory/*.txt	Corrisponde a tutti `.txt` gli oggetti a qualsiasi profondità con il prefisso s3://bucket/directory/. For example, matches s3://bucket/directory/abc.txt or s3://bucket/directory/subDir/123.txt ecc.	
s3://bucket/directory/**/*.txt	Corrisponde a tutti `.txt` gli oggetti a qualsiasi profondità con il prefisso s3://bucket/directory/. For example, matches s3://bucket/directory/abc.txt or s3://bucket/directory/subDir/123.txt ecc.	In S3, `**` è equivalente a. `*`
s3://bucket/directory/{a,b}.txt	s3://bucket/directory/a.txt, s3://bucket/directory/b.txt	
s3://bucket/directory/?. testo	Corrisponde agli oggetti nella radice del prefisso il cui nome di file è un singolo carattere seguito da. `.txt` Ad esempio, corrisponde a s3://.txt bucket/directory/a.txt but not s3://bucket/directory/someDir/a.txt or s3://bucket/directory/someDir/subDir/a	
s3://bucket/directory/[0-9].txt	s3:///9.txt bucket/directory/0.txt, s3://bucket/directory/1.txt, ... ,s3://bucket/directory	

Modello Glob	HealthOmics Comportamento della partita	Note
s3://bucket/directory/[0-9].txt	s3:///3.txt bucket/directory/1 .txt, s3://bucket/directory/2.txt, s3://bucket/directory	
s3://bucket/directory/[0-9].txt	s3://bucket/directory/b.txt, s3:// bucket/directory/c.txt, ... ,s3:// bucket/directory/Y.txt	

Gestione specifica della doppia barra negli input di Amazon S3 in base alla lingua

HealthOmics mantiene il comportamento del motore nativo per ogni motore di flusso di lavoro durante la gestione delle doppie barre in Amazon S3 URIs, in modo da non dover apportare modifiche ai flussi di lavoro durante la migrazione verso. HealthOmics Le sezioni seguenti descrivono come ogni motore gestisce diversi scenari.

WDL

Se il parametro di input include una doppia barra al centro o alla fine dell'URI, il motore WDL mantiene la doppia barra.

Parametro di input	Ubicazione prevista	
s3://1.fastq myfiles/runs/input s//file	s3://1. fastq myfiles/runs/input s//file	
s3:////myfiles/runs/inputs	s3:////myfiles/runs/inputs	

Flusso successivo

Se il parametro di input include una doppia barra al centro dell'URI, il motore Nextflow mantiene la doppia barra. Per una doppia barra alla fine dell'URI, il motore Nextflow la risolve in una singola barra.

Parametro di input	Ubicazione prevista	
s3://1.fastq myfiles/runs/inputs//file	s3://1. fastq myfiles/runs/inputs//file	
s3://myfiles//runs/inputs//*.gz	s3://myfiles//runs/inputs//*.gz	
s3://myfiles//runs/inputs//	s3://myfiles//runs/inputs/	

CWL

Se il parametro di input include una doppia barra al centro o alla fine dell'URI, il motore CWL mantiene la doppia barra.

Parametro di input	Ubicazione prevista	
s3://myfiles// runs/inputs//file 1.fastq	s3://myfiles// 1.fastq runs/inputs//file	
s3://myfiles//runs/inputs//	s3://myfiles//runs/inputs//	

Stati di archiviazione degli input di Amazon S3

HealthOmics può recuperare gli oggetti Amazon S3 che S3 fornisce in tempo reale. Per gli oggetti che si trovano nei seguenti stati di archiviazione archiviati, restore gli oggetti a cui renderli disponibili: HealthOmics

- Classi di storage Flexible Retrieval o Deep Archive in Amazon S3 Glacier.
- Livelli Archived Access o Deep Archive Access in Intelligent tiering.

Per informazioni sul ripristino degli oggetti, consulta Restoring an archived object nella Amazon S3 User Guide.

Esegui il ciclo di vita in un flusso di lavoro HealthOmics

È possibile tenere traccia dell'avanzamento di una corsa monitorando lo stato dell'esecuzione. HealthOmics aggiorna lo stato di esecuzione man mano che l'esecuzione procede nel suo ciclo di vita.

È possibile recuperare lo stato di esecuzione utilizzando uno dei seguenti metodi:

- La HealthOmics console visualizza lo stato di ogni esecuzione sulla Runs pagina.
- L'operazione GetRun API restituisce lo stato di esecuzione corrente.
- È possibile monitorare lo stato dell'esecuzione utilizzando EventBridge gli eventi. Per ulteriori informazioni, consulta Utilizzo EventBridge con AWS HealthOmics.

Argomenti

Valori dello stato di esecuzione

Quando si avvia una corsa, HealthOmics imposta lo stato dell'esecuzione suPending. Man mano che l'esecuzione procede nel suo ciclo di vita, HealthOmics aggiorna il valore dello stato in base all'avanzamento corrente.

> **Note**
>
> Non sono previsti addebiti durante uno stato di esecuzione diverso da In esecuzione. Per ulteriori informazioni, consulta la prossima sezione.

HealthOmics supporta i seguenti valori di stato di esecuzione:

Pending (In attesa)

La corsa è in coda, in attesa di inizio. Le esecuzioni in genere rimangono in sospeso per un breve periodo prima di iniziare.

- Le esecuzioni possono rimanere in sospeso più a lungo se si inviano più lavori contemporaneamente.
- Le esecuzioni rimangono in sospeso dopo che l'account ha raggiunto il numero massimo di esecuzioni simultanee.
- Un'esecuzione rimane in sospeso se fa parte di un gruppo di esecuzioni che ha raggiunto uno dei valori massimi relativi alle risorse.
- È possibile modificare le priorità delle esecuzioni in modo che le esecuzioni specifiche in coda inizino prima delle altre. Per ulteriori informazioni sulla priorità di esecuzione, vedere. Priorità di esecuzione

Avvio in corso

HealthOmics crea l'esecuzione e fornisce le risorse necessarie per l'esecuzione (ad esempio lo storage temporaneo e il nodo del motore).

- HealthOmics fornisce lo storage temporaneo di esecuzione all'inizio dell'esecuzione e defornisce lo storage di esecuzione quando l'esecuzione è in fase di arresto.

In esecuzione

Un'esecuzione rimane nello stato In esecuzione durante il processo di importazione, l'elaborazione di ogni attività e il processo di esportazione.

- HealthOmics importa i file di input nel file system temporaneo di archiviazione di esecuzione. I file di input sono di sola lettura, per evitare che le attività modifichino gli input in altre attività di un flusso di lavoro.
- Durante l'esportazione dei file, HealthOmics esporta i file di output dal file system run storage alla posizione S3.
- HealthOmics invia i registri di esecuzione e i registri delle attività CloudWatch in tempo reale mentre lo stato di esecuzione è In esecuzione. Per ulteriori informazioni, consulta Effettua il login CloudWatch.

In arresto

Dopo il completamento del processo di esportazione, l'esecuzione passa allo stato Arresto.

- HealthOmics defornisce tutte le risorse (inclusi il file system run storage e il nodo del motore).

Completato

L'esecuzione passa a Completed dopo aver HealthOmics completato il deprovisioning delle risorse.

- HealthOmics ha completato tutte le attività di esecuzione ed esportato i dati di output senza errori.
- Gli output di esecuzione sono disponibili nella posizione di output URI di Amazon S3 specificata. Per WDL e CWL, HealthOmics genera un file di riepilogo dell'output di esecuzione, che fornisce informazioni su. HealthOmics esegui uscite
- I registri del manifesto di esecuzione finale e i registri del motore (se applicabili) sono disponibili in. CloudWatch
- Per le esecuzioni che supportano nuovi tentativi di attività, un'esecuzione con lo stato Completata può includere una o più attività non riuscite. Se un nuovo tentativo di operazione ha avuto esito positivo per ogni operazione non riuscita, HealthOmics passa l'esecuzione a Completata. HealthOmics assegna un nuovo ID di attività a ogni nuovo tentativo, in modo che l'esecuzione includa l'attività IDs relativa ai tentativi falliti e al tentativo completato.

Non riuscito

HealthOmics ha rilevato uno o più errori e non è riuscito a completare tutte le attività di esecuzione.

- Un'esecuzione non riuscita passa allo stato di arresto mentre HealthOmics depredispone le risorse.

Annullato

Un utente ha avviato una richiesta di annullamento dell'esecuzione.

- HealthOmics interrompe tutte le attività in esecuzione e predispone tutte le risorse.
- HealthOmics non esporta alcun dato di output di esecuzione quando un utente annulla un'esecuzione. Non hai accesso a nessun file intermedio per un'esecuzione annullata.
- All'account vengono addebitati i costi per le attività e le risorse utilizzate dall'esecuzione durante lo stato In esecuzione prima dell'annullamento.
- Non ci sono costi se si annulla un'esecuzione con lo stato In sospeso o In corso.

Ritentativi di attività

HealthOmics supporta nuovi tentativi di attività per attività che non riescono a causa di errori di servizio (codici di stato HTTP 5XX).

Se alla fine tutte le attività in esecuzione vengono completate, anche se sono stati necessari nuovi tentativi, l'esecuzione HealthOmics passa a Completata. HealthOmics assegna un nuovo ID di attività

a ogni nuovo tentativo, in modo che l'esecuzione includa l'attività IDs relativa ai tentativi falliti e al tentativo completato.

Il comportamento predefinito dei nuovi tentativi dipende dal linguaggio di definizione utilizzato dal flusso di lavoro. L'impostazione predefinita per Nextflow non prevede nuovi tentativi. Per WDL e CWL, HealthOmics tenta fino a due nuovi tentativi di un'operazione non riuscita, ma è possibile disattivare il nuovo tentativo per attività specifiche o per tutte le attività di un flusso di lavoro. Un nuovo tentativo di operazione è utile per risolvere gli errori di servizio intermittenti. Tuttavia, potresti prendere in considerazione la possibilità di rinunciare a un'attività idempotente.

Per informazioni specifiche su ciascun linguaggio di definizione del flusso di lavoro, consulta i seguenti argomenti:

- WDL — Configura il comportamento di ripetizione delle attività nella definizione del flusso di lavoro. Vedere Configurazione del comportamento di ripetizione delle attività WDL.
- Nextflow: configura il comportamento di ripetizione delle attività nel file di configurazione di Nextflow o nella definizione del flusso di lavoro. Vedi Configurare il comportamento di ripetizione delle attività di Nextflow.
- CWL — Configura il comportamento di ripetizione delle attività nella definizione del flusso di lavoro. Vedi Configurazione del comportamento di ripetizione delle attività CWL.

Implicazioni relative ai prezzi dello stato di esecuzione

Il tuo account può incorrere in addebiti mentre lo stato di esecuzione è In esecuzione. Non sono previsti addebiti durante nessun altro stato di esecuzione. Ad esempio, non è previsto alcun addebito per le risorse quando la corsa è in corso o interrotta.

Un'esecuzione con lo stato In esecuzione ha le seguenti implicazioni di fatturazione:

- All'account vengono addebitati costi per l'utilizzo del file system Run Storage mentre lo stato di esecuzione è In esecuzione. Per informazioni sui tipi di run storage, vedere. Esegui tipi di storage nei flussi HealthOmics di lavoro
- All'account vengono addebitati costi per l'esecuzione delle attività, in base alle risorse di calcolo e di memoria specificate per ciascuna attività nella definizione del flusso di lavoro e in base alla durata dell'attività. Per ulteriori informazioni, consulta Requisiti di calcolo e memoria per le attività HealthOmics.
- Ogni attività ha una soglia di fatturazione minima di un minuto. Se esegui un'attività per meno di un minuto, ti verrà addebitato un costo per almeno un minuto di utilizzo. Se possibile, raggruppa

piccole attività per ottimizzare i costi. Il raggruppamento delle attività riduce anche i tempi di esecuzione evitando la creazione di più attività sequenziali.

Per ulteriori informazioni sui HealthOmics prezzi, consulta la pagina Prezzi. HealthOmics

HealthOmics esegui uscite

Al termine di un'esecuzione WDL o CWL, gli output includono un file di riepilogo dell'output (in formato JSON) che elenca tutti gli output prodotti dall'esecuzione. È possibile utilizzare il file di riepilogo dell'output per i seguenti scopi:

- Determina a livello di codice i file di output generati dall'esecuzione.
- Verifica che l'esecuzione abbia prodotto tutti gli output previsti.

Argomenti

Esegui il riepilogo dell'output per WDL

Al termine di un'esecuzione WDL, HealthOmics crea un file di riepilogo di output denominato. output.json

Per ogni output del flusso di lavoro, nel file è presente una key/value coppia corrispondente. La chiave contiene il nome del flusso di lavoro e il nome dell'output nel seguente formato:`WorkflowName.output_name`. Per l'output di un file, il valore è un URI S3 che punta alla posizione di output in S3 in cui è archiviato il file. Per un output Array [File], il valore è un array di S3. URIs

L'esempio seguente mostra il output.json file per un flusso di lavoro denominatoBWAMappingWorkflow.

```
{
  "BWAMappingWorkflow.bam_indexes": [
    "s3://omics-outputs/8886192/out/bam_indexes/0/
pbmc8k_S1_L007_R1_001.sorted.bam.bai",
    "s3://omics-outputs/8886192/out/bam_indexes/1/pbmc8k_S1_L008_R1_001.sorted.bam.bai"
  ],
```

```
  "BWAMappingWorkflow.mapping_stats": "s3://omics-outputs/8886192/out/mapping_stats/
genome_mapping_final_stats.txt",
  "BWAMappingWorkflow.merged_bam": "s3://omics-outputs/8886192/out/merged_bam/
genome_mapping.merged.bam",
  "BWAMappingWorkflow.merged_bam_index": "s3://omics-outputs/8886192/out/
merged_bam_index/genome_mapping.merged.bam.bai",
  "BWAMappingWorkflow.reference_index_tar": "s3://omics-outputs/8886192/out/
reference_index_tar/reference_index.tar",
  "BWAMappingWorkflow.sorted_bams": [
    "s3://omics-outputs/8886192/out/sorted_bams/0/pbmc8k_S1_L007_R1_001.sorted.bam",
    "s3://omics-outputs/8886192/out/sorted_bams/1/pbmc8k_S1_L008_R1_001.sorted.bam"
  ],
  "BWAMappingWorkflow.unmapped_bams": [
    "s3://omics-outputs/8886192/out/unmapped_bams/0/
pbmc8k_S1_L007_R1_001.unmapped.bam",
    "s3://omics-outputs/8886192/out/unmapped_bams/1/pbmc8k_S1_L008_R1_001.unmapped.bam"
  ]
}
```

Se il flusso di lavoro produce output con tipi diversi da file (come String, Int, Float o Bool), il valore del campo è una primitiva JSON. Per esempio:

```
{
  "MyWorkflow.my_int_ouput": 1,
  "MyWorkflow.my_bool_output": false,
    ...
}
```

Esegui il riepilogo dell'output per CWL

Al termine di un'esecuzione CWL, HealthOmics crea un file di riepilogo dell'output denominato outputs.json nella seguente posizione:

```
{my-S3outputpath}/{runId}/{run-uuid}/logs/outputs.json
```

Il file di riepilogo dell'output include un elenco di output. Ogni uscita è una key/value coppia, dove la chiave è il nome dell'output. Il valore è un oggetto che include le seguenti proprietà:

- location: il percorso completo del file di output
- basename — La parte del percorso relativa al nome del file
- class — Il tipo di output, che in genere è File

- size — La dimensione del file in byte

Nell'esempio seguente, il file output.json ha un elenco di due file di output.

```
{
  "example_output": {
    "location": "{my-S3outputpath}/{runId}/{run-uuid}/out/output.txt",
    "basename": "output.txt",
    "class": "File",
    "size": 13
  },
  "another_output": {
    "location": "{my-S3outputpath}/{runId}/{run-uuid}/out/metrics.json",
    "basename": "metrics.json",
    "class": "File",
    "size": 256
  }
}
```

Motivi dell'errore di esecuzione

Se un'esecuzione non riesce, utilizza l'operazione GetRunAPI per recuperare il motivo dell'errore.

Esamina il motivo dell'errore per aiutarti a risolvere il motivo per cui l'esecuzione non è riuscita. La tabella seguente elenca ogni motivo di errore insieme a una descrizione dell'errore.

Motivo dell'errore	Descrizione dell'errore.
ASSUME_ROLE_FAILED	HealthOmics non ha il permesso di assumere il ruolo. Specifica re il HealthOmics principale nella relazione di fiducia per il ruolo.
ERRORE CANNOT_ST ART_CONTAINER_	Impossibile avviare l'attività di workflow:, id: container utilizzan do image:*name*. *ID image name* Assicurati che l'immagine sia valida e riprova.
CANNOT_START_CONTA INER_SIZE_ERROR	Impossibile avviare l'operazione di workflow:, id: container utilizzando image:. *name ID image name* Assicurati che la dimensione dell'immagine sia inferiore a 45 GiB (95 GiB per un'istanza GPU) e riprova.

Motivo dell'errore	Descrizione dell'errore.
ECR_PERMISSION_ERROR	HealthOmics non dispone dell'autorizzazione per accedere all'URI dell'immagine. Verifica che l'archivio privato di Amazon ECR esista e che abbia concesso l'accesso al HealthOmics servizio principale.
ESPORTAZIONE_FALLITA	Esportazione non riuscita. Verifica che il bucket di output esista e che il ruolo di esecuzione disponga dell'autorizzazione di scrittura per il bucket.
FILE_SYSTEM_OUT_OF_SPACE	Il file system non dispone di spazio sufficiente. Aumenta le dimensioni del file system ed esegui di nuovo.
ERRORE DI VERIFICA DELL'IMMAGINE	Impossibile verificare l'immagine. *`image name`* Per correggere il problema, prova a estrarre l'immagine e inseriscila nuovamente nell'archivio ECR.
IMPORTAZIONE_FALLITA	L'importazione non è riuscita. Verifica che il file di input esista e che il ruolo di esecuzione possa accedere all'input.
INACTIVE_OMICS_STORAGE_RESOURCE	L'URI di archiviazione non è in stato ACTIVE. HealthOmics Attiva il set di lettura e riprova. Per ulteriori informazioni sull'attivazione dei set di lettura, consultaAttivazione dei set di lettura in HealthOmics.
INPUT_URI_NOT_FOUND	L'URI fornito non esiste:. *`uri`* Verifica che il percorso URI esista e conferma che il ruolo possa accedere all'oggetto.
INSTANCE_RESERVATION_FAILED	La capacità dell'istanza non è sufficiente per completare l'esecuzione del flusso di lavoro. Attendi e riprova a eseguire il flusso di lavoro.
INVALID_ECR_IMAGE_URI	La struttura URI dell'immagine Amazon ECR non è valida. Fornisci un URI valido e riprova.
VALORE_TASK_RESOURCE_VALUE NON VALIDO	La GPU, la CPU o la memoria richieste sono troppo elevate per la capacità di elaborazione disponibile o sono inferiori al valore minimo di 1 per l'attività. *`ID`*

Motivo dell'errore	Descrizione dell'errore.
INPUT URI NON VALIDO	La struttura URI non è valida. *uri* Controlla la struttura dell'URI e riprova.
MODIFIED_INPUT_RESOURCE	L'URI *uri* fornito è stato modificato dopo l'inizio dell'esecuzione. Riprova l'esecuzione.
OUT_OF_MEMORY_ERROR	L'attività del flusso di lavoro ha esaurito la memoria. *ID* Aumenta il valore di memoria nella definizione del flusso di lavoro e riprova a eseguire.
RUN_TASK_FAILED	L'esecuzione non è riuscita perché l'operazione non è riuscita. Per eseguire il debug dell'errore dell'attività, utilizza l'operazione GetRunTaskAPI e il flusso Amazon CloudWatch Logs.
RUN_TIMED_OUT	Timeout di esecuzione dopo pochi minuti. *number*
ERRO_SERVIZIO	Si è verificato un errore temporaneo nel servizio. Prova a eseguire nuovamente il flusso di lavoro.
TASK_TIMED_OUT	Attività scaduta dopo pochi secondi. *id number*
UNSUPPORTED_INPUT_SIZE	La dimensione totale dell'input è troppo alta. Riduci la dimension e di input e riprova.
WORKFLOW_RUN_FAILED	Esecuzione del workflow non riuscita. Esamina il flusso di log del motore CloudWatch Logs: *ID* per eseguire il debug dell'errore.
WORKFLOW_VER_VALIDATION_FAILED	HealthOmics non supporta la versione di Nextflow richiesta: --. *version* L'ultima versione supportata è. *version* Modifica la tua versione di Nextflow con una versione supportata e riprova.
UNSUPPORTED_GPU_INSTANCE_TYPE	Il tipo di istanza richiesto non è supportato in. *Region* Riprova l'esecuzione con un tipo di istanza GPU supportato in questa regione. I tipi di istanza disponibili sono. *GPU instance types*

Guida per le esecuzioni che non rispondono

Durante lo sviluppo di nuovi flussi di lavoro, esecuzioni o attività specifiche potrebbero «bloccarsi» o «bloccarsi» se ci sono problemi con il codice e le attività non riescono a uscire correttamente dai processi. Questo può essere difficile da risolvere e da catturare, poiché è normale che le attività vengano eseguite per periodi prolungati. Per prevenire e identificare esecuzioni che non rispondono, segui le migliori pratiche suggerite nelle sezioni seguenti.

Le migliori pratiche per prevenire le esecuzioni che non rispondono

- Assicurati di chiudere tutti i file aperti nel codice dell'attività. L'apertura di troppi file può occasionalmente causare problemi di threading all'interno dei motori di flusso di lavoro.
- I processi in background creati da un'attività del flusso di lavoro devono terminare quando l'operazione viene chiusa. Tuttavia, se un processo in background non viene chiuso correttamente, è necessario chiuderlo in modo esplicito nel codice dell'attività.
- Assicuratevi che i processi non si ripetano senza uscire. Ciò può causare una mancata risposta dell'esecuzione e la risoluzione richiede una modifica al codice di definizione del flusso di lavoro.
- Fornisci un'allocazione di memoria e CPU appropriata per le tue attività. Analizza CloudWatch i log o utilizza le esecuzioni Esegui Analyzer completate con successo del flusso di lavoro per verificare di disporre di un'allocazione di elaborazione ottimale. Utilizzate il `headroom` parametro Run Analyzer per includere un margine di crescita aggiuntivo, assicurandovi che i processi dispongano di risorse sufficienti per il completamento. Includi almeno il 5% di margine di crescita nella memoria e nella CPU allocate, per tenere conto dei processi del sistema operativo in background.
  - Inoltre, aumenta la dimensione della larghezza di banda dell'istanza se l'istanza richiede un throughput più elevato. EC2 Le istanze Amazon con meno di 16 v CPUs (dimensioni 4xl e inferiori) possono subire un aumento del throughput. Per ulteriori informazioni sulla velocità effettiva delle EC2 istanze Amazon, consulta Larghezza di banda EC2 disponibile delle istanze Amazon.
- Assicurati di utilizzare la dimensione corretta del file system per le tue esecuzioni. Per le esecuzioni che non rispondono che utilizzano lo storage a esecuzione statica, è consigliabile aumentare l'allocazione dello storage di esecuzione statica per consentire una maggiore velocità di I/O e una maggiore capacità di archiviazione sul file system. Analizza il run manifest per vedere lo storage massimo del file system, utilizza Run Analyzer per determinare se è necessario aumentare l'allocazione del file system.

Le migliori pratiche per rilevare le esecuzioni che non rispondono

- Quando sviluppi nuovi flussi di lavoro, usa un gruppo di esecuzione con il limite massimo di tempo di esecuzione impostato su catch runaway code. Ad esempio, se il completamento di una corsa richiede 1 ora, inseriscila in un gruppo di corsa che scada dopo 2 o 3 ore (o un periodo di tempo diverso in base al caso d'uso) per catturare lavori in fase di esecuzione. Inoltre, applica un buffer per tenere conto della varianza nei tempi di elaborazione.
- Imposta una serie di gruppi di esecuzione con limiti di runtime massimi diversi. Ad esempio, è possibile assegnare esecuzioni brevi a un gruppo di esecuzione che termina le esecuzioni dopo alcune ore e a un gruppo di esecuzioni lunghe che termina le esecuzioni dopo alcuni giorni, in base alla durata prevista del flusso di lavoro.
- HealthOmics ha un limite di durata massima del servizio predefinito di 604.800 secondi, o 7 giorni, che è regolabile tramite una richiesta nello strumento per le quote. Richiedi un aumento del limite di servizio di questa quota solo se hai esecuzioni della durata di circa una settimana. Se disponi di una combinazione di tirature brevi e lunghe e non utilizzi i gruppi di corsa, valuta la possibilità di inserire le esecuzioni di lunga durata in un account separato con un limite di durata massima del servizio più elevato.
- Esamina CloudWatch i registri per individuare eventuali attività che ritieni non rispondano. Se un'attività normalmente genera istruzioni di registro regolari e non lo fa da un periodo prolungato, è probabile che l'attività sia bloccata o bloccata.

Cosa fare se si verifica un'esecuzione che non risponde

- Annulla la corsa per evitare di incorrere in costi aggiuntivi.
- Controlla i registri delle attività per verificare se alcuni processi non sono riusciti a terminare correttamente.
- Ispezionate i registri del motore per identificare eventuali comportamenti anomali del motore.
- Confrontate i registri delle attività e del motore dell'esecuzione che non risponde con quelli delle esecuzioni identiche completate con successo. Questo può aiutare a identificare eventuali differenze che potrebbero aver causato il comportamento di mancata risposta.
- Se non riesci a determinare la causa principale, richiedi assistenza e includi quanto segue:
  - ARN dell'esecuzione bloccata e ARN di un'esecuzione identica completata con successo.
  - Registri del motore (disponibili dopo che la corsa è stata annullata o fallita)
  - Registri delle attività per l'attività che non risponde. Per la risoluzione dei problemi non sono necessari i registri delle attività per tutte le attività del flusso di lavoro.

Ciclo di vita delle attività in esecuzione HealthOmics

Un'attività è un singolo processo all'interno di un'esecuzione. HealthOmics mappa ogni attività del flusso di lavoro su un tipo di istanza di calcolo omico che meglio si adatta alle risorse richieste dall'attività. Le risorse richieste vengono specificate nella definizione del flusso di lavoro. Per ulteriori informazioni, vedereRequisiti di calcolo e memoria per le attività HealthOmics.

HealthOmics fornisce una memoria di esecuzione temporanea per l'attività da utilizzare. HealthOmics copia i file di input dell'operazione nell'archivio temporaneo di esecuzione come file di sola lettura. HealthOmics fornisce collegamenti simbolici in modo che l'attività possa accedere ai file di input dalla directory di lavoro. L'attività ha accesso solo ai file dichiarati nel file di definizione del flusso di lavoro.

Valori dello stato dell'attività

È possibile tenere traccia dell'avanzamento di un'attività monitorandone lo stato. Quando si avvia un'esecuzione, HealthOmics imposta lo stato dell'attività su Pending ogni attività in esecuzione. Quando l'attività viene avviata e prosegue nel suo ciclo di vita, HealthOmics aggiorna il valore dello stato in base all'avanzamento corrente.

È possibile recuperare lo stato dell'attività utilizzando uno dei seguenti metodi:

- La HealthOmics console visualizza lo stato di ogni attività in esecuzione sulla Run details pagina.
- L'operazione GetRunTask API restituisce lo stato dell'attività.
- È possibile monitorare lo stato delle attività utilizzando EventBridge gli eventi. Per ulteriori informazioni, consulta Utilizzo EventBridge con AWS HealthOmics.

È possibile recuperare lo stato corrente di un'attività utilizzando l'operazione GetRunTask API. La HealthOmics console visualizza lo stato di ogni attività in esecuzione sulla Run details pagina.

HealthOmics supporta i seguenti valori di stato delle attività:

In attesa

L'operazione è in coda, in attesa di essere avviata. Le attività rimangono in sospeso per un breve periodo prima di iniziare.

- Le attività rimangono in sospeso dopo che l'account ha raggiunto il numero massimo di attività simultanee.
- Le attività rimangono in sospeso se l'esecuzione fa parte di un gruppo di esecuzione che ha raggiunto uno dei valori massimi di risorse.

- È possibile modificare le priorità di esecuzione in modo che le esecuzioni specifiche in coda e le relative attività vengano avviate prima delle altre esecuzioni in coda. Per ulteriori informazioni sulla priorità di esecuzione, vedere Priorità di esecuzione

Avvio in corso

HealthOmics sta creando l'attività e fornendo le risorse necessarie per l'attività, ad esempio il nodo delle attività del flusso di lavoro.

In esecuzione

Lo stato dell'operazione è In esecuzione durante HealthOmics l'elaborazione dell'operazione.

In arresto

Dopo aver completato l'operazione di elaborazione ed esportazione dei dati di output, l'operazione passa a Interruzione.

- HealthOmics depredispone il nodo dell'attività del flusso di lavoro.

Completato

HealthOmics ha terminato l'elaborazione dell'operazione e ha trasferito i dati di output nel file system run storage.

Non riuscito

HealthOmics ha riscontrato un errore durante l'elaborazione dell'operazione e non l'ha completata.

- L'attività passa allo stato Interruzione (rifornisce HealthOmics le risorse) e quindi allo stato Non riuscito.
- Se l'errore è un errore di servizio (codice di stato HTTP 5XX) e il flusso di lavoro supporta nuovi HealthOmics tentativi per questa operazione, tenta di elaborare nuovamente l'operazione. HealthOmics assegna un nuovo ID di attività al nuovo tentativo.

Annullato

HealthOmics interrompe l'operazione dopo una richiesta di annullamento dell'esecuzione avviata dall'utente.

- L'attività passa allo stato Interruzione (rifornisce le risorse) e HealthOmics quindi allo stato Annullato.

Risoluzione dei problemi delle attività del flusso

Di seguito sono riportate le best practice e le considerazioni per la risoluzione dei problemi relativi alle attività.

- I log delle attività si basano sull'attività `STDOUT` e `STDERR` vengono prodotti da essa. Se l'applicazione utilizzata nell'attività non produce nessuno di questi, non ci sarà un registro delle attività. Per facilitare il debug, utilizzate le applicazioni in modalità. `verbose`
- Per visualizzare i comandi eseguiti in un'operazione insieme ai relativi valori interpolati, utilizzate il comando Bash. `set -x` Questo può aiutare a determinare se l'attività sta utilizzando gli input corretti e a identificare dove gli errori potrebbero aver impedito l'esecuzione dell'attività come previsto.
- Utilizzate il `echo` comando per inviare i valori delle variabili a `STDOUT` o`STDERR`. Questo ti aiuta a confermare che siano impostate come previsto.
- Usa comandi come `ls -l <name_of_input_file>` per confermare che gli input siano presenti e abbiano le dimensioni previste. Se non lo sono, ciò potrebbe rivelare un problema con un'attività precedente che produceva output vuoti a causa di un bug.
- Utilizzate il comando `df -Ph . | awk 'NR==2 {print $4}'` in uno script di task per determinare lo spazio attualmente disponibile per l'attività e aiutare a identificare le situazioni in cui potrebbe essere necessario eseguire il flusso di lavoro con un'allocazione di storage aggiuntiva.

L'inclusione di uno qualsiasi dei comandi precedenti in uno script di attività presuppone che il contenitore delle attività includa anche questi comandi e che si trovino nell'ambiente `path` del contenitore.

Esegui l'ottimizzazione per un HealthOmics flusso di lavoro privato

Puoi ottimizzare le esecuzioni per il costo totale, il tempo di esecuzione totale o una combinazione di entrambi. HealthOmics fornisce dati e strumenti per aiutarvi a prendere decisioni di ottimizzazione delle esecuzioni. L'ottimizzazione delle esecuzioni non si applica ai flussi di lavoro Ready2Run, poiché non hai alcun controllo sul modo in cui il servizio gestisce l'approvvigionamento delle risorse per questi flussi di lavoro.

Il primo passaggio consiste nel comprendere l'utilizzo corrente delle risorse e il costo delle attività in esecuzione, quindi applicare metodi per ottimizzare i costi e le prestazioni di esecuzione.

Argomenti

- Esegui Analyzer
- Determina i costi di esecuzione
- Determina l'utilizzo in fase di esecuzione
- Metodi per ottimizzare le esecuzioni
- Impatto della variazione delle dimensioni dei file tra le esecuzioni
- Metodi per ottimizzare la concorrenza delle risorse

Esegui Analyzer

HealthOmics fornisce uno strumento open source denominato Run Analyzer. Questo strumento estrae informazioni sull'utilizzo delle risorse a livello di attività per una corsa e suggerisce opportunità di ottimizzazione dei costi e delle prestazioni di esecuzione.

> **Note**
>
> Run analyzer stima i costi delle attività e i potenziali risparmi sui costi in base ai prezzi di AWS listino al momento dell'esecuzione dello strumento. Valuta i consigli di ottimizzazione e implementa quelli più adatti ai tuoi casi d'uso. Testa le ottimizzazioni che adotti per assicurarti che funzionino per la tua corsa.

Run Analyzer esegue le seguenti attività:

- Valuta i colli di bottiglia di memoria e calcolo.
- Identifica le attività che richiedono un eccessivo approvvigionamento di memoria o CPU e consiglia nuove dimensioni delle istanze in grado di ridurre i costi.
- Calcola le stime dei costi per le singole attività e calcola i potenziali risparmi sui costi se si applicano i consigli.
- Fornisce una visualizzazione cronologica delle attività in modo da poter verificare le dipendenze tra le attività e la sequenza di elaborazione. La sequenza temporale consente inoltre di identificare le attività di lunga durata.
- Fornisce consigli sulla dimensione del file system per l'archiviazione di esecuzione.
- Mostra i tempi di approvvigionamento delle attività in modo da poter identificare le aree in cui grandi carichi di container potrebbero rallentare i tempi di approvvigionamento.

- Lo strumento include un parametro di input (headroom) che puoi utilizzare per controllare l'aggressività dei consigli di ottimizzazione.

Le sezioni seguenti includono suggerimenti specifici per l'utilizzo di Run Analyzer per ottimizzare le esecuzioni.

Determina i costi di esecuzione

È possibile utilizzare i seguenti metodi e linee guida per determinare i costi di gestione:

- Per visualizzare i costi di gestione totali per un periodo di fatturazione, procedi nel seguente modo:

  1. Apri la console Billing and Cost Management e scegli Bills.
  2. In Charges by service, espandi Omics.
  3. Espandi la regione, quindi visualizza il costo di tutte le tue esecuzioni suddivise per tipo di istanza omics, tipo di storage di esecuzione e flusso di lavoro Ready2Run.

- Per generare un rapporto sui costi che includa informazioni per ogni esecuzione, procedi nel seguente modo:

  1. Apri la console Billing and Cost Management e scegli Esportazioni dati.
  2. Scegli Crea per creare una nuova esportazione di dati.
  3. Inserisci un nome di esportazione per l'esportazione dei dati. Mantieni gli altri campi ai valori predefiniti per creare un rapporto CUR (costo e utilizzo).
  4. Per la granularità temporale, seleziona ogni ora o ogni giorno.
  5. In Impostazioni di archiviazione per l'esportazione dei dati, esegui questi passaggi di configurazione:

     a. Configura un bucket Amazon S3 per l'esportazione dei dati.

     b. Per il controllo delle versioni dei file, seleziona se sovrascrivere il file di esportazione esistente o creare ogni volta un nuovo file.

     Il sistema genera il primo rapporto entro le 24 ore successive e genera i report successivi una volta al giorno.
  6. Per ulteriori informazioni su come creare l'esportazione dei dati, vedere Creazione di esportazioni di dati nella Guida per l'utente delle esportazioni di AWS dati.

- Puoi etichettare le tue tirature per monitorare e ottimizzare i costi per categoria, ad esempio per team o per progetto. Se utilizzi i tag, segui questi passaggi per visualizzare i costi di esecuzione per categoria di tag:

  1. Apri la console Billing and Cost Management e scegli Cost Explorer.
  2. In Parametri del rapporto > Raggruppa per, scegli Tag come dimensione e seleziona il nome del tag desiderato.
- Per vedere l'utilizzo delle risorse per le attività, visualizza i log in del manifesto di esecuzione. CloudWatch Per ulteriori informazioni, consulta Monitoraggio HealthOmics con CloudWatch registri.
- Utilizzate lo Esegui Analyzer strumento per estrarre le informazioni sull'utilizzo delle risorse delle attività per un'esecuzione.

Determina l'utilizzo in fase di esecuzione

È possibile utilizzare i seguenti metodi per analizzare l'utilizzo in fase di esecuzione:

- Dalla pagina Esecuzioni della console, è possibile visualizzare il tempo di esecuzione totale di un'esecuzione.
- Dalla pagina dei dettagli dell'esecuzione, è possibile visualizzare i seguenti elementi:
  - Visualizza il tempo di esecuzione totale di una corsa.
  - Visualizza il tempo di esecuzione per ogni attività in esecuzione.
  - Scegli uno dei link per visualizzare i log in Amazon S3 o per visualizzare i log di esecuzione o i log in di esecuzione del manifest. CloudWatch
- Dall'elenco Esegui attività, scegli il link Visualizza registri relativo a un'attività per visualizzare i log delle attività. CloudWatch
- La risposta all'operazione `listRuns` API include l'ora di inizio e l'ora di fine dell'esecuzione, in modo da poter calcolare il tempo di esecuzione totale.
- Lo Esegui Analyzer strumento mostra la durata delle attività in una visualizzazione temporale. Questo strumento fornisce una rappresentazione visiva della sequenza di elaborazione delle attività, che è possibile abbinare all'ordine previsto.

Metodi per ottimizzare le esecuzioni

HealthOmics fornisce, gestisce e ottimizza automaticamente le risorse che eseguono l'archiviazione temporanea dei dati (ad esempio importazioni ed esportazioni di dati). HealthOmics inoltre avvia

ed esegue il motore di workflow relativo al tuo workflow. Tuttavia, è possibile influenzare gli orari di inizio dell'esecuzione, gli orari di inizio delle attività e il tempo di esecuzione complessivo dell'attività impostando varie configurazioni di esecuzione. L'approccio generale alla definizione e alla progettazione del flusso di lavoro influisce anche sul tempo di esecuzione delle attività. L'elenco seguente descrive i fattori che possono influire sull'esecuzione e sulle prestazioni delle attività:

Esegui il tipo di archiviazione

Il tipo di run storage ha un impatto sulle prestazioni di esecuzione e sul tempo di esecuzione del provisioning. Il Dynamic Run Storage esegue il provisioning più velocemente e non esaurisce mai la memoria, perché si adatta dinamicamente alle esigenze di run storage. Lo storage a esecuzione dinamica è ideale anche per i flussi di lavoro in fase di sviluppo, in cui spesso è possibile avviare e arrestare un flusso di lavoro per risolvere i problemi.

Lo storage a esecuzione statica richiede tempi di provisioning del file system più lunghi, ma può completare alcune esecuzioni più velocemente, in genere se le esecuzioni hanno un'elevata concomitanza delle attività o richiedono una capacità del file system superiore a 9,6 TiB. Lo storage a esecuzione statica è ideale per flussi di lavoro di lunga durata con requisiti elevati. I/O

Per aiutarti a valutare il rapporto costo/prestazioni di ogni tipo di run storage per una determinata esecuzione, puoi provare i test A/B per vedere quale tipo di run storage offre prestazioni migliori. Inoltre, prendi in considerazione l'utilizzo di Dynamic Run Storage per i tuoi cicli di sviluppo, quindi utilizza lo storage a esecuzione statica per esecuzioni di produzione su larga scala.

Per ulteriori informazioni sui tipi di run storage Esegui tipi di storage nei flussi HealthOmics di lavoro

Esegui un provisioning eccessivo dello storage statico

Se il calcolo delle attività del flusso di lavoro è limitato da I/O, consider over-provisioning the static run storage. Storage cost increases with its size, but maximum throughput of the file system also increases. If an expensive compute task is experiencing I/O colli di bottiglia, l'aumento delle dimensioni del file system per ridurre il tempo di esecuzione delle attività può ridurre il costo complessivo.

Riduci le dimensioni delle immagini dei contenitori

All'avvio di ogni attività, HealthOmics carica il contenitore specificato per l'attività. I contenitori più grandi richiedono più tempo per essere caricati. Ottimizza i contenitori in modo che siano il più piccoli possibile per migliorare l'efficienza nell'avvio di nuove attività. Se aggiungi set di dati di grandi dimensioni ai tuoi contenitori, valuta la possibilità di archiviare i set di dati in S3 e di

fare in modo che il flusso di lavoro importi i dati da S3. Per le dimensioni massime dei contenitori supportate, consulta HealthOmics . HealthOmics quote a dimensione fissa per il flusso di lavoro

Dimensioni processo

È possibile combinare piccole attività sequenziali in un'unica attività per ridurre i tempi di assegnazione delle attività. Inoltre, HealthOmics prevede una tariffa minima per la durata delle attività di un minuto, quindi la combinazione delle attività può ridurre i costi. Nell'ambito dell'operazione combinata, potresti essere in grado di utilizzare le pipe Unix per evitare i I/O costi di serializzazione e deserializzazione dei file.

Compressione dei file

Evita di comprimere eccessivamente i file intermedi del flusso di lavoro. La maggior parte dei formati genomici utilizza la compressione «gzip» o «block gzip». La decompressione del file di input dell'attività e la ricompressione del file di output dell'attività possono consumare una grande percentuale dell'utilizzo complessivo della CPU dell'operazione. Alcune applicazioni di genomica consentono di impostare il livello di compressione durante la serializzazione degli output. Riducendo il livello di compressione, è possibile ridurre il tempo impiegato dalla CPU, sebbene file più grandi aumentino il tempo impiegato per la scrittura su disco. A seconda dell'attività e dell'applicazione, è possibile individuare il livello di compressione ottimale per i file intermedi che garantiscono il tempo di esecuzione più breve. Si consiglia di iniziare indirizzando le attività con i file di output più grandi. Un livello di compressione pari a 2 è ideale per diversi scenari. Puoi iniziare con questo livello in base al tuo caso d'uso e confrontare i risultati provando altri livelli di compressione.

Numero di thread

Se specifichi i thread nella definizione dell'attività, imposta il numero di thread sullo stesso valore del numero di v richiesti. CPUs

Specificate calcolo e memoria

Se non specificate risorse di memoria o di calcolo nell'attività, HealthOmics assegna il tipo di istanza più piccolo (`omics.c.large`) come predefinito. Dichiara esplicitamente i tuoi requisiti di memoria e calcolo se desideri HealthOmics assegnare un tipo di istanza più grande.

HealthOmics alloca il numero di risorse vCPUs, memoria e GPU richieste. Ad esempio, se richiedi 15v CPUs e 33GiB, HealthOmics alloca un'istanza omics.m.4xl (16v, 64GB) per l'attivitàCPUs, ma l'attività può utilizzare solo 15 v e 33GiB. CPUs Pertanto, ti consigliamo di richiedere risorse v e di memoria che corrispondano a un'istanza omics. CPUs

Batch più campioni in un'unica esecuzione

Poiché il provisioning del file system richiede tempo all'inizio dell'esecuzione, è possibile risparmiare tempo di provisioning raggruppando più campioni nella stessa esecuzione. Considerate i seguenti fattori prima di scegliere questo approccio:

- Un singolo campione errato può causare il fallimento di un flusso di lavoro, pertanto la suddivisione in batch dei campioni potrebbe aumentare il numero di flussi di lavoro non riusciti. Se non sei sicuro che il tuo flusso di lavoro avrà successo per la maggior parte del tempo, un'esecuzione per campione potrebbe essere un approccio migliore.
- HealthOmics alloca un unico file system di archiviazione per l'intero flusso di lavoro. Per un batch di campioni, assicuratevi di specificare una quantità sufficiente di storage di esecuzione per elaborare tutti i campioni.
- È prevista una quantità massima di storage di esecuzione per flusso di lavoro, pertanto ciò potrebbe limitare il numero di campioni che è possibile aggiungere al batch.
- La dimensione minima dello storage di esecuzione è di 1,2 TiB, quindi il batch può ridurre i costi se il flusso di lavoro utilizza molto meno spazio di archiviazione rispetto al minimo per ogni campione.
- Run storage è in grado di gestire più connessioni simultanee, quindi avere più attività che utilizzano lo stesso storage di esecuzione non dovrebbe causare colli di bottiglia. I/O
- Ogni esecuzione ha il proprio set di tag. Se contrassegni i flussi di lavoro con informazioni per la definizione del budget o il monitoraggio, potrebbe essere preferibile utilizzare esecuzioni separate.
- I ruoli IAM si applicano all'intera esecuzione. Ogni utente ha accesso a tutti i dati per un batch di campioni. La separazione dei flussi di lavoro consente di utilizzare autorizzazioni più dettagliate.
- HealthOmics imposta quote a livello di account per il numero massimo di flussi di lavoro simultanei e il numero massimo di attività simultanee in un flusso di lavoro. Per informazioni su come richiedere un aumento di queste quote, vedere. HealthOmics quote di servizio

Usa i parametri per le immagini dei contenitori

Parametrizza le immagini del contenitore anziché incorporarle nel flusso di lavoro URIs . Se si tratta di parametri di esecuzione, HealthOmics verifica che l'esecuzione abbia accesso ai contenitori prima dell'inizio dell'esecuzione. In caso contrario, l'attività ha esito negativo durante l'esecuzione, quando sono stati sostenuti addebiti per tutte le attività completate. Inoltre, poiché si tratta di input con parametri, HealthOmics genera un checksum nel manifesto di esecuzione, che migliora la provenienza dell'esecuzione.

Usa un linter

Utilizzate un linter per trovare gli errori più comuni del flusso di lavoro prima di eseguire un nuovo flusso di lavoro. Per ulteriori informazioni, consulta Stampanti per flussi di lavoro in HealthOmics.

Utilizzatelo EventBridge per segnalare i problemi

Utilizza avvisi EventBridge personalizzati per rilevare anomalie specifiche della tua logica aziendale.

Usa gli archivi di sequenza

Prendi in considerazione l'utilizzo di un archivio di sequenze per i dati di origine per risparmiare sui costi di archiviazione. Per ulteriori informazioni, consulta Store omics data in modo conveniente su qualsiasi scala con HealthOmics post sul blog.

Impatto della variazione delle dimensioni dei file tra le esecuzioni

Gli utenti spesso progettano e testano le esecuzioni utilizzando un piccolo set di dati di test, quindi utilizzano un'ampia varietà di dati con variazioni significative delle dimensioni dei file nei cicli di produzione. Assicurati di tenere conto di questa varianza quando ottimizzi l'esecuzione.

L'elenco seguente descrive i consigli per l'ottimizzazione in caso di variazioni significative nelle dimensioni dei file:

Variate le dimensioni dei file nei dati di test

Prova a utilizzare dati di test durante lo sviluppo che abbiano una varianza rappresentativa.

Usa Run Analyzer

Utilizza lo strumento Run Analyzer su una varietà di campioni per tenere conto della varianza nelle dimensioni dei dati.

È possibile utilizzare lo strumento Run Analyzer per comprendere la varianza tra le esecuzioni nei campioni di dati di produzione. Utilizzate `--batch` la modalità di Run Analyzer per generare statistiche per un batch di esecuzioni e analizzare le risorse di calcolo massime necessarie per gestire i valori anomali nei set di dati.

Ad esempio, è possibile assegnare a run analyzer una cella di dati a flusso completo in modalità batch per comprendere il picco di utilizzo della vCPU e della memoria per la cella a flusso completo.

Ridurre la varianza delle dimensioni dei set di dati di input

Se si riscontra un'elevata varianza nelle dimensioni dei campioni, è possibile biforcare i campioni a monte del file system HealthOmics e selezionare diverse dimensioni del file system per ogni batch per risparmiare sui costi di storage di esercizio.

In WDL, utilizzate la `size` funzione per biforcare l'allocazione delle risorse per singole attività per campioni grandi rispetto a campioni piccoli. Applica questa strategia alle attività più costose per avere il massimo impatto.

In Nextflow, utilizza risorse condizionali per suddividere l'allocazione delle risorse in base alla dimensione o al nome del file. Per ulteriori informazioni, consulta Risorse di processo condizionali sul sito Nextflow. GitHub

Non ottimizzate troppo presto

Finalizza il codice e la logica del flusso di lavoro prima di investire in importanti sforzi di ottimizzazione delle prestazioni. La modifica del codice può avere un impatto significativo sulle risorse richieste. Se ottimizzi un'esecuzione troppo presto nel processo di sviluppo, potresti ottimizzarla eccessivamente o potrebbe essere necessario ottimizzarla nuovamente se la definizione del flusso di lavoro cambia in un secondo momento.

Eseguite nuovamente lo strumento Run Analyzer periodicamente

Se apporti modifiche alla definizione del flusso di lavoro nel tempo o se la varianza del campione cambia, esegui periodicamente lo strumento Run Analyzer per apportare ottimizzazioni aggiuntive.

Metodi per ottimizzare la concorrenza delle risorse

HealthOmics offre le seguenti funzionalità per aiutarvi a controllare e gestire i costi quando l'elaborazione viene eseguita su larga scala:

- Utilizza i gruppi di esecuzione per controllare i costi e l'utilizzo delle risorse. È possibile impostare valori massimi nel gruppo di esecuzione per il numero di esecuzioni simultanee CPUs GPUs, v e il tempo di esecuzione totale per attività. Se team o gruppi diversi utilizzano lo stesso account, puoi creare un gruppo di corsa separato per ogni team. Puoi controllare l'utilizzo delle risorse e i costi per team e configurando i valori massimi del gruppo di esecuzione. Per ulteriori informazioni, consulta Utilizzo dei gruppi di HealthOmics esecuzione.

- Durante lo sviluppo, puoi configurare un gruppo di esecuzione separato con valori massimi inferiori per catch task in esecuzione.
- Le Service Quotas aiutano anche a proteggere il tuo account da richieste eccessive di risorse. Per informazioni su Service Quotas, incluso come richiedere aumenti del valore delle quote, vedere HealthOmics quote di servizio

Esegui operazioni in HealthOmics

Puoi avviare, rieseguire, clonare, annullare o eliminare una corsa:

- Start— HealthOmics crea una nuova esecuzione utilizzando le impostazioni di configurazione specificate e quindi avvia l'esecuzione.
- Rerun— HealthOmics crea una nuova esecuzione che è un duplicato dell'esecuzione specificata. È possibile eseguire nuovamente una corsa eliminata utilizzando lo strumento. HealthOmics rerun
- Clone— È possibile clonare un'esecuzione esistente utilizzando la console. La console apre la pagina Clone run e precompila i campi di configurazione utilizzando i valori dell'esecuzione esistente. È possibile modificare i valori in base alle esigenze e avviare l'esecuzione clonata.
- Cancel— È possibile annullare un'esecuzione non ancora completata. Quando si annulla una corsa, HealthOmics non salva nessuno degli output di esecuzione.
- Delete— È possibile eliminare manualmente le esecuzioni completate oppure impostare la modalità di conservazione delle esecuzioni per HealthOmics eliminare automaticamente le esecuzioni più vecchie. Per ulteriori informazioni sulla modalità di conservazione, vederethe section called “Esegui le modalità di conservazione”.

Argomenti

- Inizia una corsa in HealthOmics
- Esegui nuovamente un run in HealthOmics
- Clona un run in HealthOmics
- Annullare un run in HealthOmics
- Eliminare un run in HealthOmics

Inizia una corsa in HealthOmics

Quando si avvia un'esecuzione, si specificano le risorse da HealthOmics allocare per l'uso durante l'esecuzione.

Specificare il tipo di archiviazione di esecuzione e la quantità di archiviazione (per l'archiviazione statica). Per garantire l'isolamento e la sicurezza dei dati, HealthOmics effettua il provisioning dello storage all'inizio di ogni esecuzione e lo disattiva alla fine dell'esecuzione. Per ulteriori informazioni, consulta Esegui tipi di storage nei flussi HealthOmics di lavoro.

Specificare una posizione Amazon S3 per i file di output. Se esegui contemporaneamente un volume elevato di flussi di lavoro, utilizza un URIs output Amazon S3 separato per ogni flusso di lavoro per evitare il bucket throttling. Per ulteriori informazioni, consulta Organization object using prefixes nella Amazon S3 User Guide e Scalare orizzontalmente le connessioni di storage nel white paper Optimizing Amazon S3 Performance.

Puoi anche specificare la priorità di esecuzione. L'impatto della priorità sulla corsa dipende dal fatto che la corsa sia associata a un gruppo di esecuzioni. Per ulteriori informazioni, consulta Priorità di esecuzione.

Se un flusso di lavoro ha una o più versioni, puoi specificare una versione all'avvio dell'esecuzione. Se non si specifica una versione, HealthOmics avvia la versione predefinita del flusso di lavoro.

Quando si utilizza l' HealthOmics API, è possibile fornire un ID di richiesta univoco per ogni esecuzione. L'ID della richiesta è un token di idempotenza che viene HealthOmics utilizzato per identificare le richieste duplicate e avvia l'esecuzione una sola volta.

> **Note**
>
> Si specifica un ruolo del servizio IAM quando si avvia un'esecuzione. Facoltativamente, la console può creare il ruolo di servizio per te. Per ulteriori informazioni, consulta Ruoli di servizio per AWS HealthOmics.

Argomenti

HealthOmics parametri di esecuzione

Quando si avvia un'esecuzione, si specificano gli input di esecuzione nel file JSON dei parametri di esecuzione oppure è possibile immettere i valori dei parametri in linea. Per informazioni sulla gestione delle dimensioni del file JSON dei parametri di esecuzione, vedere. Gestione delle dimensioni dei parametri di esecuzione

HealthOmics supporta i seguenti tipi JSON per i valori dei parametri.

tipo JSON	Chiave e valore di esempio	Note
booleano	«b»: vero	Il valore non è tra virgolette ed è tutto in minuscolo.
integer	«i»: 7	Il valore non è tra virgolette.
number	«f»: 42,3	Il valore non è tra virgolette.
string	«s» :"caratteri»	Il valore è tra virgolette. Usa il tipo di stringa per i valori di testo e URIs. La destinazione URI deve essere il tipo di input previsto.
array	«a»: [1,2,3]	Il valore non è tra virgolette. Ciascun membro dell'array deve avere il tipo definito dal parametro di input.
oggetto	«o»: {"left» :"a», «right» :1}	In WDL, gli oggetti vengono mappati su WDL Pair, Map o Struct

Avvio di una corsa utilizzando la console

Per iniziare una corsa

1. Apri la HealthOmics console.
2. Se necessario, apri il pannello di navigazione a sinistra (≡). Scegli Runs.

3. Nella pagina Esecuzioni, scegli Avvia esecuzione.
4. Nel pannello dei dettagli dell'esecuzione, fornisci le seguenti informazioni

   - Fonte del flusso di lavoro: scegli Flusso di lavoro proprietario o Flusso di lavoro condiviso.
   - ID del flusso di lavoro: l'ID del flusso di lavoro associato a questa esecuzione.
   - Versione del flusso di lavoro (opzionale): seleziona una versione del flusso di lavoro da utilizzare per questa esecuzione. Se non si seleziona una versione, l'esecuzione utilizza la versione predefinita del flusso di lavoro.
   - Nome esecuzione: nome distintivo per questa esecuzione.
   - Priorità di esecuzione (facoltativa): la priorità di questa esecuzione. I numeri più alti indicano una priorità più alta e le attività con la priorità più alta vengono eseguite per prime.
   - Esegui tipo di archiviazione: specifica qui il tipo di archiviazione per sostituire il tipo di archiviazione di esecuzione predefinito specificato per il flusso di lavoro. Lo storage statico alloca una quantità fissa di spazio di archiviazione per l'esecuzione. Lo storage dinamico è scalabile verso l'alto e verso il basso in base alle esigenze di ogni attività in esecuzione.
   - Run storage: per lo storage di esecuzione statico, specifica la quantità di storage necessaria per l'esecuzione. Questa voce sostituisce la quantità di storage di esecuzione predefinita specificata per il flusso di lavoro.
   - Seleziona la destinazione di output S3: la posizione S3 in cui verranno salvati gli output di esecuzione.
   - ID dell'account del proprietario del bucket di output (opzionale): se il tuo account non possiede il bucket di output, inserisci l'ID del proprietario del bucket. Account AWS Queste informazioni sono necessarie per HealthOmics verificare la proprietà del bucket.
   - Esegui la modalità di conservazione dei metadati: scegli se conservare i metadati per tutte le esecuzioni o se fare in modo che il sistema rimuova i metadati di esecuzione più vecchi quando l'account raggiunge il numero massimo di esecuzioni. Per ulteriori informazioni, consulta Esegui la modalità di conservazione per le HealthOmics esecuzioni.
5. In Ruolo di servizio, puoi utilizzare un ruolo di servizio esistente o crearne uno nuovo.
6. (Facoltativo) Per i tag, puoi assegnare fino a 50 tag alla corsa.
7. Scegli Next (Successivo).
8. Nella pagina Aggiungi valori dei parametri, fornisci i parametri di esecuzione. Puoi caricare un file JSON che specifica i parametri o inserire manualmente i valori.
9. Scegli Next (Successivo).

10. Nel pannello Esegui gruppo, puoi facoltativamente specificare un gruppo di esecuzione per questa esecuzione. Per ulteriori informazioni, consulta Utilizzo dei gruppi di HealthOmics esecuzione.
11. Nel pannello Esegui cache, puoi facoltativamente specificare una cache di esecuzione per questa esecuzione. Per ulteriori informazioni, consulta Configurazione di un run with run cache utilizzando la console.
12. Selezionare Review and start run (Verifica e avvia sessione).
13. Dopo aver esaminato la configurazione di esecuzione, scegliete Avvia esecuzione.

Avvio di un'esecuzione utilizzando l'API

Utilizza l'operazione API start-run per creare e avviare un'esecuzione.

L'esempio seguente specifica l'ID del flusso di lavoro e il ruolo del servizio. Questo esempio imposta la modalità di conservazione su`REMOVE`. Per ulteriori informazioni sulla modalità di conservazione, vedereEsegui la modalità di conservazione per le HealthOmics esecuzioni.

```
aws omics start-run
     --workflow-id workflow id \
     --role-arn arn:aws:iam::1234567892012:role/service-role/
OmicsWorkflow-20221004T164236 \
     --name workflow name \
     --retention-mode REMOVE
```

In risposta, si ottiene il seguente risultato. `uuid`È unico per l'esecuzione e `outputUri` può essere utilizzato insieme per tenere traccia di dove vengono scritti i dati di output.

```
{
    "arn": "arn:aws:omics:us-west-2:....:run/1234567",
    "id": "123456789",
    "uuid":"96c57683-74bf-9d6d-ae7e-f09b097db14a",
    "outputUri":"s3://bucket/folder/8405154/96c57683-74bf-9d6d-ae7e-f09b097db14a"
    "status": "PENDING"
}
```

Include un file di parametri

Se il modello di parametri per un flusso di lavoro dichiara i parametri richiesti, è possibile fornire un file JSON locale degli input quando si avvia l'esecuzione di un flusso di lavoro. Il file JSON contiene il nome esatto di ogni parametro di input e un valore per il parametro.

Fai riferimento al file JSON di input in AWS CLI aggiungendolo `--parameters file://<input_file.json>` alla tua `start-run` richiesta. Per ulteriori informazioni sui parametri di esecuzione, vedereHealthOmics eseguire ingressi.

Fornire un ID di richiesta

Puoi fornire un nome univoco `requestId` per ogni corsa. L'ID della richiesta è un token di idempotenza che viene utilizzato HealthOmics per catturare richieste duplicate. Non avvierà un'esecuzione se l'ID della richiesta è un duplicato di un'esecuzione precedente.

Se utilizzi l'infrastruttura (come le funzioni Lambda o le funzioni step) per orchestrare gli avvii di esecuzione, la migliore pratica consiste nel fornire un ID di richiesta univoco per ogni richiesta. StartRun Ciò garantisce che se l'infrastruttura avvia inavvertitamente un'esecuzione già iniziata, HealthOmics non avvii l'esecuzione duplicata. Ad esempio, se l'infrastruttura sta tentando di ripristinare un errore a monte, può eseguire nuovamente uno script che tenta di avviare esecuzioni che sono richieste duplicate.

Scegli una versione del flusso di lavoro

È possibile specificare una versione del flusso di lavoro per l'esecuzione. Se non si specifica una versione, HealthOmics avvia l'esecuzione con la versione del flusso di lavoro predefinita.

```
aws omics start-run
     --workflow-id workflow id \
      ...
     --workflow-version-name '1.2.1'
```

Sostituisci il tipo di archiviazione di esecuzione

È possibile sovrascrivere il tipo di run storage predefinito impostato nel flusso di lavoro.

```
aws omics start-run
       --workflow-id workflow id \
        ...
       --storage-type STATIC
```

```
       --storage-capacity 2400
```

Esegui un flusso di lavoro GPU

Puoi anche specificare un ID di workflow GPU, come illustrato nell'esempio seguente:

```
aws omics start-run
       --workflow-id workflow id \
       --role-arn arn:aws:iam::1234567892012:role/service-role/
OmicsWorkflow-20221004T164236 \
       --name GPUTestRunModel \
       --output-uri s3://amzn-s3-demo-bucket1
```

Ottieni informazioni su una corsa

Puoi utilizzare l'ID nella risposta con l'API get-run per verificare lo stato di un'esecuzione, come mostrato.

```
aws omics get-run --id run id
```

La risposta di questa operazione API indica lo stato dell'esecuzione del flusso di lavoro. Gli stati possibili sonoPENDING, STARTINGRUNNING, eCOMPLETED. Quando c'è un'esecuzioneCOMPLETED, puoi trovare un file di output chiamato `outfile.txt` nel tuo bucket Amazon S3 di output, in una cartella che prende il nome dall'ID di esecuzione.

L'operazione API get-run restituisce anche altri dettagli, ad esempio se il flusso di lavoro è `Ready2Run` o`PRIVATE`, il motore del flusso di lavoro e i dettagli dell'acceleratore. L'esempio seguente mostra la risposta per get-run for a run di un workflow privato, descritta in WDL con un acceleratore GPU e nessun tag assegnato all'esecuzione.

```
{
    "arn": "arn:aws:omics:us-west-2:123456789012:run/7830534",
    "id": "7830534",
    "uuid":"96c57683-74bf-9d6d-ae7e-f09b097db14a",
    "outputUri":"s3://bucket/folder/8405154/96c57683-74bf-9d6d-ae7e-f09b097db14a"
    "status": "COMPLETED",
    "workflowId": "4074992",
    "workflowType": "PRIVATE",
    "workflowVersionName": "3.0.0",
    "roleArn": "arn:aws:iam::123456789012:role/service-role/
OmicsWorkflow-20221004T164236",
```

```
    "name": "RunGroupMaxGpuTest",
    "runGroupId": "9938959",
    "digest":
 "sha256:a23a6fc54040d36784206234c02147302ab8658bed89860a86976048f6cad5ac",
    "accelerators": "GPU",
    "outputUri": "s3://amzn-s3-demo-bucket1",
    "startedBy": "arn:aws:sts::123456789012:assumed-role/Admin/<role_name>",
    "creationTime": "2023-04-07T16:44:22.262471+00:00",
    "startTime": "2023-04-07T16:56:12.504000+00:00",
    "stopTime": "2023-04-07T17:22:29.908813+00:00",
    "tags": {}
}
```

È possibile visualizzare lo stato di tutte le esecuzioni con l'operazione API list-runs, come illustrato.

```
aws omics list-runs
```

Per visualizzare tutte le attività completate per un'esecuzione specifica, utilizza l'list-run-tasksAPI.

```
aws omics list-run-tasks --id task ID
```

Per ottenere i dettagli di un'attività specifica, utilizza l' get-run-taskAPI.

```
aws omics get-run-task --id <run_id> --task-id task ID
```

Al termine dell'esecuzione, i metadati vengono inviati a CloudWatch under the stream. **`manifest/run/<run ID>/<run UUID>`**

Di seguito è riportato un esempio del manifesto.

```
{
    "arn": "arn:aws:omics:us-east-1:123456789012:run/1695324",
    "creationTime": "2022-08-24T19:53:55.284Z",
    "resourceDigests": {
      "s3://omics-data/broad-references/hg38/v0/Homo_sapiens_assembly38.dict":
 "etag:3884c62eb0e53fa92459ed9bff133ae6",
      "s3://omics-data/broad-references/hg38/v0/Homo_sapiens_assembly38.fasta":
 "etag:e307d81c605fb91b7720a08f00276842-388",
      "s3://omics-data/broad-references/hg38/v0/Homo_sapiens_assembly38.fasta.fai":
 "etag:f76371b113734a56cde236bc0372de0a",
      "s3://omics-data/intervals/hg38-mjs-whole-chr.500M.intervals":
 "etag:27fdd1341246896721ec49a46a575334",
```

```
      "s3://omics-data/workflow-input-lists/dragen-gvcf-list.txt":
 "etag:e22f5aeed0b350a66696d8ffae453227"
    },
    "digest":
 "sha256:a5baaff84dd54085eb03f78766b0a367e93439486bc3f67de42bb38b93304964",
    "engine": "WDL",
    "main": "gatk4-basic-joint-genotyping-v2.wdl",
    "name": "1044-gvcfs",
    "outputUri": "s3://omics-data/workflow-output",
    "parameters": {
      "callset_name": "cohort",
      "input_gvcf_uris": "s3://omics-data/workflow-input-lists/dragen-gvcf-list.txt",
      "interval_list": "s3://omics-data/intervals/hg38-mjs-whole-chr.500M.intervals",
      "ref_dict": "s3://omics-data/broad-references/hg38/v0/
Homo_sapiens_assembly38.dict",
      "ref_fasta": "s3://omics-data/broad-references/hg38/v0/
Homo_sapiens_assembly38.fasta",
      "ref_fasta_index": "s3://omics-data/broad-references/hg38/v0/
Homo_sapiens_assembly38.fasta.fai"
    },
    "roleArn": "arn:aws:iam::123456789012:role/OmicsServiceRole",
    "startedBy": "arn:aws:sts::123456789012:assumed-role/admin/ahenroid-Isengard",
    "startTime": "2022-08-24T20:08:22.582Z",
    "status": "COMPLETED",
    "stopTime": "2022-08-24T20:08:22.582Z",
    "storageCapacity": 9600,
    "uuid": "a3b0ca7e-9597-4ecc-94a4-6ed45481aeab",
    "workflow": "arn:aws:omics:us-east-1:123456789012:workflow/1558364",
    "workflowType": "PRIVATE"
  },
  {
    "arn": "arn:aws:omics:us-east-1:123456789012:task/1245938",
    "cpus": 16,
    "creationTime": "2022-08-24T20:06:32.971290",
    "image": "123456789012.dkr.ecr.us-west-2.amazonaws.com/gatk",
    "imageDigest":
 "sha256:8051adab0ff725e7e9c2af5997680346f3c3799b2df3785dd51d4abdd3da747b",
    "memory": 32,
    "name": "geno-123",
    "run": "arn:aws:omics:us-east-1:123456789012:run/1695324",
    "startTime": "2022-08-24T20:08:22.278Z",
    "status": "SUCCESS",
    "stopTime": "2022-08-24T20:08:22.278Z",
    "uuid": "44c1a30a-4eee-426d-88ea-1af403858f76"
```

```
  },
  ...
```

I metadati di esecuzione non vengono eliminati se non sono presenti nei CloudWatch registri.

Esegui nuovamente un run in HealthOmics

Per le esecuzioni che non hai ancora eliminato, usa la console o l'API per rieseguirle. Per le esecuzioni che hai eliminato, usa lo strumento. HealthOmics rerun

Argomenti

- Esegui nuovamente un'esecuzione utilizzando la console
- Esegui nuovamente un'esecuzione utilizzando l'API
- Utilizzo dello strumento Rerun

Esegui nuovamente un'esecuzione utilizzando la console

Dalla console, segui questi passaggi per eseguire nuovamente un'esecuzione:

1. Apri la HealthOmics console.
2. Se necessario, apri il pannello di navigazione a sinistra (≡). Scegli Runs.
3. Nella pagina Esecuzioni, seleziona l'esecuzione da rieseguire.
4. Dal menu delle azioni sopra la tabella, scegli Riesegui.

Esegui nuovamente un'esecuzione utilizzando l'API

Utilizza l'operazione StartRun API per rieseguire un'esecuzione esistente. Fornisci i seguenti input richiesti:

- Un ruolo di servizio ARN (`roleArn`).
- L'ID della corsa da duplicare (`runId`).
- Una posizione Amazon S3 in cui l'esecuzione salva gli output di esecuzione (). `outputUri`

```
aws omics start-run
     --run-id run id \
     --role-arn arn:aws:iam::1234567892012:role/service-role/
OmicsWorkflow-20221004T164236 \
```

```
    --output-uri s3://workflow-output-b6f2fce1
```

Utilizzo dello strumento Rerun

Per un'esecuzione eliminata, è possibile scaricare e utilizzare HealthOmics rerun lo strumento per eseguire nuovamente l'esecuzione. Lo strumento recupera le informazioni sull'esecuzione dal manifesto Logs. CloudWatch Scaricate lo rerun strumento dal repository HealthOmics Tool GitHub .

L'esempio seguente mostra come utilizzare lo rerun strumento.

```
aws-healthomics-rerun 9876543
```

Se l'esecuzione esiste in CloudWatch, si riceve una risposta simile all'output di esempio seguente. Se il flusso di lavoro non esiste più, viene visualizzato un messaggio di errore.

```
Original request:
{
  "workflowId": "9679729",
  "roleArn": "arn:aws:iam::123456789012:role/DemoRole",
  "name": "sample_rerun",
  "parameters": {
    "image": "123456789012.dkr.ecr.us-west-2.amazonaws.com/default:latest",
    "file1": "omics://123456789012.storage.us-west-2.amazonaws.com/8647780323/
readSet/6389608538"
  },
  "outputUri": "s3://workflow-output-bcf2fcb1"
}
StartRun request:
{
  "workflowId": "9679729",
  "roleArn": "arn:aws:iam::123456789012:role/DemoRole",
  "name": "new test",
  "parameters": {
    "image": "123456789012.dkr.ecr.us-west-2.amazonaws.com/default:latest",
    "file1": "omics://123456789012.storage.us-west-2.amazonaws.com/8647780323/
readSet/6389608538"
  },
  "outputUri": "s3://workflow-output-bcf2fcb1"
}
StartRun response:
{
  "arn": "arn:aws:omics:us-west-2:123456789012:run/9171779",
```

```
  "id": "9171779",
  "status": "PENDING",
  "tags": {}
}
```

Clona un run in HealthOmics

Puoi clonare un'esecuzione esistente utilizzando la HealthOmics console. La clonazione crea una nuova esecuzione utilizzando i valori di configurazione della corsa clonata. È possibile modificare questi valori predefiniti e aggiungere altri input opzionali.

1. Apri la HealthOmics console.
2. Se necessario, apri il pannello di navigazione a sinistra (≡). Scegli Runs.
3. Nella pagina Esecuzioni, seleziona l'esecuzione da clonare.
4. Dal menu delle azioni sopra la tabella, scegli Clone run. La console apre il modulo Clone run. Il modulo è identico a Start run, tranne per il fatto che la console compila il modulo con tutti i valori pertinenti dell'esecuzione clonata.

   La console crea un nuovo ID di esecuzione per il clone di esecuzione e aggiunge questo ID di esecuzione come suffisso al nome dell'esecuzione.

   Man mano che procedi attraverso le pagine del modulo, puoi modificare i valori di configurazione secondo necessità.
5. Dopo aver esaminato la configurazione di esecuzione, scegli Avvia esecuzione.

Annullare un run in HealthOmics

Puoi annullare una corsa se il suo stato è `PENDING`,`STARTING`,`RUNNING`, o`STOPPING`.

> **Note**
>
> Quando si annulla una corsa, HealthOmics non salva nessuno degli output di esecuzione.

Dalla console, segui questi passaggi per annullare un'esecuzione:

1. Apri la HealthOmics console.
2. Se necessario, apri il pannello di navigazione a sinistra (≡). Scegli Runs.

3. Nella pagina Esecuzioni, scegli la corsa da annullare.
4. La console apre la pagina dei dettagli dell'esecuzione. Dal banner di stato nella parte superiore della pagina, scegli Stop run.
5. Inserisci conferma per interrompere la corsa.

Per annullare un'esecuzione utilizzando l'API, utilizza l'operazione CancelRun API.

L'esempio seguente mostra come annullare un'esecuzione utilizzando AWS CLI . Per eseguire l'esempio, sostituisci il *run id* con l'ID della corsa che desideri annullare. In caso di successo, non viene fornita alcuna risposta.

```
aws omics cancel-run --id run id
```

Eliminare un run in HealthOmics

Quando non ti serve più un'esecuzione, puoi eliminarla utilizzando l' AWS CLI API o la console. Puoi eliminare un'esecuzione quando il suo stato è `COMPLETED` o`CANCELED`.

Dalla console, segui questi passaggi per eliminare un'esecuzione:

1. Apri la HealthOmics console.
2. Se necessario, apri il pannello di navigazione a sinistra (≡). Scegli Runs.
3. Nella pagina Esecuzioni, seleziona una o più esecuzioni da eliminare.
4. Dal menu delle azioni sopra la tabella, scegli Elimina.
5. Nel modulo modale, digita confirm per confermare l'eliminazione.

Il AWS CLI comando seguente elimina una corsa. Per eseguire l'esempio, sostituisci *run id* con l'ID della corsa che desideri eliminare. Se la corsa viene eliminata correttamente, non viene fornita alcuna risposta.

```
aws omics delete-run --id run id
```

Utilizzo dei gruppi di HealthOmics esecuzione

Facoltativamente, puoi creare un gruppo di esecuzione per limitare le risorse di calcolo per le esecuzioni che aggiungi al gruppo. I gruppi Run possono aiutarti a:

- Metti in coda le tue esecuzioni in modo da non superare i limiti di servizio.
- Rileva le attività in corso di esecuzione impostando una durata massima di esecuzione.
- Gestisci la priorità di ogni esecuzione in modo che le esecuzioni più importanti vengano completate per prime.

Se si imposta il numero massimo di vCPU, GPU o esecuzioni simultanee, le attività di esecuzione verranno messe in coda quando viene raggiunto il numero massimo. Se si imposta una durata massima di esecuzione, l'esecuzione ha esito negativo se supera la durata massima.

Utilizzate l'impostazione della priorità di esecuzione per stabilire la priorità all'interno di un gruppo di esecuzione.

I limiti di servizio hanno la precedenza sui limiti del gruppo di esecuzione. Ad esempio, se si imposta un numero massimo di run group su un valore superiore a quello massimo del servizio in una regione, HealthOmics applica il valore massimo del servizio.

Argomenti

Priorità di esecuzione

È possibile utilizzare la priorità di esecuzione per stabilire la priorità delle esecuzioni in un gruppo di esecuzione.

Se più esecuzioni hanno la stessa priorità, l'esecuzione iniziata per prima ha la priorità più alta.

Puoi anche impostare una priorità per una corsa che non fa parte di un gruppo di corse. La priorità viene confrontata con le priorità di tutte le altre esecuzioni che non fanno parte di un gruppo di corse

La priorità di esecuzione viene impostata all'inizio della corsa. Per ulteriori informazioni, consulta Inizia una corsa in HealthOmics.

Crea un gruppo di corsa utilizzando la console

Per creare un gruppo di corsa

1. Apri la HealthOmics console.
2. Se necessario, apri il pannello di navigazione a sinistra (≡). Scegli Esegui gruppi.
3. Nella pagina Esegui gruppi, scegli Crea gruppo di corsa.
4. Nella pagina dei dettagli di Crea gruppo di corsa, fornisci le seguenti informazioni

   - Nome del gruppo di esecuzione: un nome univoco per questo gruppo di esecuzione.
   - Numero massimo di vCPU per esecuzioni simultanee: il numero massimo di v CPUs che possono essere eseguite contemporaneamente su tutte le esecuzioni attive nel gruppo di esecuzione.
   - Max GPUs: il numero massimo di esecuzioni GPUs che possono essere eseguite contemporaneamente su tutte le esecuzioni attive del gruppo di esecuzione.
   - Tempo di esecuzione massimo (minuti) per esecuzione: il tempo massimo per ogni esecuzione (in minuti). Se un'esecuzione supera il tempo di esecuzione massimo, l'esecuzione ha esito negativo automaticamente.
   - Numero massimo di esecuzioni simultanee: il numero massimo di esecuzioni che possono essere eseguite contemporaneamente.
5. (opzionale) Puoi aggiungere fino a 50 tag al gruppo di esecuzione.
6. Scegli Crea gruppo di corsa.

Crea un gruppo di esecuzione utilizzando la CLI

Per creare un gruppo di esecuzione, utilizzate l'operazione create-run-groupAPI per creare un gruppo di esecuzione denominato`TestRunGroup`. L'esempio seguente imposta un massimo di 20 CPUs, 10 GPUs, 5 esecuzioni e una durata massima di esecuzione di 600 minuti.

```
aws omics create-run-group --name TestRunGroup \
--max-cpus 20 \
--max-gpus 10 \
--max-duration 600 \
--max-runs 5
```

La risposta di questa operazione API include l'ID del nuovo file creato`RunGroup`.

```
{
    "arn": "arn:aws:omics:us-west-2:12345678901:runGroup/2839621",
    "id": "2839621",
    "tags": {}
}
```

Per ottenere informazioni aggiuntive sul gruppo di esecuzione, utilizzate questo ID con l'operazione get-run-groupAPI, come illustrato nell'esempio seguente.

```
aws omics get-run-group --id run group id
```

La risposta include le impostazioni dei limiti per il gruppo di esecuzione e i tag assegnati.

```
{
    "arn": "arn:aws:omics:us-west-2:776893852117:runGroup/2839621",
    "id": "2839621",
    "name": "TestRunGroup",
    "maxCpus": 20,
    "maxRuns": 5,
    "maxDuration": 600,
    "creationTime": "2024-06-12T15:35:39.191730+00:00",
    "tags": {},
    "maxGpus": 10
}
```

Puoi anche utilizzare l'operazione list-run-groupAPI per visualizzare tutti i gruppi di esecuzione creati.

```
aws omics list-run-groups
```

Eliminare un gruppo di esecuzione utilizzando la console

È possibile eliminare un gruppo di esecuzione se non vi sono esecuzioni associate a quel gruppo di esecuzione con lo stato di `PENDINGSTARTING`,`RUNNING`, o`STOPPING`.

Per eliminare un gruppo di corsa, segui questi passaggi.

1. Apri la HealthOmics console.
2. Se necessario, apri il pannello di navigazione a sinistra (≡). Scegli Esegui gruppi.
3. Nella pagina Esegui gruppi, scegli il gruppo di esecuzione da eliminare e scegli Elimina nella casella xx.

Eliminare un gruppo di esecuzione utilizzando la CLI

È possibile eliminare un gruppo di esecuzione se non vi sono esecuzioni associate a quel gruppo di esecuzione con lo stato di`PENDING`, `STARTINGRUNNING`, o`STOPPING`.

L'esempio seguente mostra come utilizzare AWS CLI per eliminare un gruppo di corsa. Non riceverai alcuna risposta. Per eseguire l'esempio, sostituisci *`run group id`* con l'ID del gruppo di esecuzione che desideri eliminare.

```
aws omics delete-run-group --id run group id
```

Memorizzazione nella cache delle chiamate per le esecuzioni HealthOmics

AWS HealthOmics supporta la memorizzazione nella cache delle chiamate, nota anche come resume, per flussi di lavoro privati. La memorizzazione nella cache delle chiamate salva gli output delle attività del flusso di lavoro completate al termine di un'esecuzione. Le esecuzioni successive possono utilizzare gli output delle attività dalla cache, anziché calcolare nuovamente gli output delle attività. La memorizzazione nella cache delle chiamate riduce l'utilizzo delle risorse di elaborazione, il che si traduce in durate di esecuzione più brevi e risparmi sui costi di elaborazione.

È possibile accedere ai file di output delle attività memorizzati nella cache al termine dell'esecuzione. Per eseguire il debug avanzato e la risoluzione dei problemi delle attività, è possibile memorizzare nella cache i file di attività intermedi specificando tali file come output delle attività nella definizione del flusso di lavoro.

È possibile utilizzare la memorizzazione nella cache delle chiamate per salvare i risultati delle attività completate dalle esecuzioni non riuscite. L'esecuzione successiva inizia dall'ultima attività completata con successo, anziché calcolare nuovamente le attività completate.

Se HealthOmics non trova una voce della cache corrispondente per un'attività, l'esecuzione non ha esito negativo. HealthOmics ricalcola l'attività e le relative attività dipendenti.

Per informazioni sulla risoluzione dei problemi relativi alla memorizzazione nella cache delle chiamate, vedere. Risoluzione dei problemi relativi alla memorizzazione nella cache delle chiamate

Argomenti

Come funziona la memorizzazione nella cache delle chiamate

Per utilizzare la memorizzazione nella cache delle chiamate, devi creare una cache di esecuzione e configurarla per avere una posizione Amazon S3 associata per i dati memorizzati nella cache. Quando avvii un'esecuzione, specifichi la cache di esecuzione. Una cache di esecuzione non è dedicata a un unico flusso di lavoro. Le esecuzioni da più flussi di lavoro possono utilizzare la stessa cache.

Durante la fase di esportazione di un'esecuzione, il sistema esporta gli output delle attività completate nella posizione Amazon S3. Per esportare file di attività intermedi, dichiarali come output delle attività nella definizione del flusso di lavoro. La memorizzazione nella cache delle chiamate salva inoltre internamente i metadati e crea hash unici per ogni voce della cache.

Per ogni attività in esecuzione, il motore di workflow rileva se esiste una voce della cache corrispondente per questa attività. Se non esiste una voce di cache corrispondente, HealthOmics calcola l'operazione. Se è presente una voce della cache corrispondente, il motore recupera i risultati memorizzati nella cache.

Per abbinare le voci della cache, HealthOmics utilizza il meccanismo di hashing incluso nei motori di flusso di lavoro nativi. HealthOmicsestende queste implementazioni hash esistenti per tenere conto di HealthOmics variabili, come S3 ETags e container digest ECR.

HealthOmics supporta la memorizzazione nella cache delle chiamate per queste versioni linguistiche del flusso di lavoro:

- versioni WDL 1.0, 1.1 e versione di sviluppo
- Nextflow versione 23.10 e 24.10
- Tutte le versioni CWL

> **Note**
>
> HealthOmics non supporta la memorizzazione nella cache delle chiamate per i flussi di lavoro Ready2Run.

Argomenti

Modello di responsabilità condivisa

Esiste una responsabilità condivisa tra gli utenti e consiste nel determinare se le attività e AWS le esecuzioni siano idonee per la memorizzazione nella cache delle chiamate. La memorizzazione nella cache delle chiamate consente di ottenere i migliori risultati quando tutte le attività sono idempotenti (le esecuzioni ripetute di un'attività utilizzando gli stessi input producono gli stessi risultati).

Tuttavia, se un'attività include elementi non deterministici (come generazioni di numeri casuali o l'ora del sistema), le esecuzioni ripetute dell'attività utilizzando gli stessi input possono generare output diversi. Ciò può influire sull'efficacia della memorizzazione nella cache delle chiamate nei seguenti modi:

- Se HealthOmics utilizza una voce della cache (creata da un'esecuzione precedente) che non è identica all'output che l'esecuzione dell'attività produrrebbe per l'esecuzione corrente, l'esecuzione potrebbe produrre risultati diversi rispetto alla stessa esecuzione senza memorizzazione nella cache.
- HealthOmics potrebbe non trovare una voce della cache corrispondente per un'attività che dovrebbe corrispondere, a causa degli output dell'attività non deterministici. Se non trova la voce di cache valida, run ricalcola inutilmente l'operazione, con conseguente riduzione dei vantaggi in termini di costi derivanti dall'utilizzo della memorizzazione nella cache delle chiamate.

Di seguito sono riportati i comportamenti noti delle attività che possono causare risultati non deterministici che influiscono sui risultati della memorizzazione nella cache delle chiamate:

- Utilizzo di generatori di numeri casuali.
- Dipendenza dall'ora del sistema.
- Utilizzo della concorrenza (le condizioni di gara possono causare variazioni nell'output).
- Recupero di file locali o remoti oltre a quanto specificato nei parametri di input dell'attività.

Per altri scenari che possono causare comportamenti non deterministici, consulta Input di processo non deterministici nel sito di documentazione di Nextflow.

Se sospetti che un'attività produca output non deterministici, prendi in considerazione l'utilizzo delle funzionalità del motore di workflow per evitare di memorizzare nella cache attività specifiche non deterministiche. Per istruzioni su come disattivare la memorizzazione nella cache per singole attività in ogni lingua del flusso di lavoro supportata, consulta. Funzionalità di memorizzazione nella cache specifiche del motore

Ti consigliamo di esaminare attentamente i requisiti specifici del flusso di lavoro e delle attività prima di abilitare la memorizzazione nella cache delle chiamate in qualsiasi ambiente in cui una memorizzazione inefficace delle chiamate o output diversi dal previsto possono comportare rischi. Ad esempio, è necessario considerare attentamente le potenziali limitazioni della memorizzazione nella cache delle chiamate per determinare se la memorizzazione nella cache delle chiamate sia appropriata per i casi d'uso clinico.

Requisiti di memorizzazione nella cache per le attività

HealthOmics memorizza nella cache gli output delle attività che soddisfano i seguenti requisiti:

- L'attività deve definire un contenitore. HealthOmics non memorizzerà nella cache gli output di un'attività senza contenitore.
- L'attività deve produrre uno o più output. Gli output delle attività vengono specificati nella definizione del flusso di lavoro.
- La definizione del flusso di lavoro non deve utilizzare valori dinamici. Ad esempio, se si passa un parametro a un'attività con un valore che aumenta ad ogni esecuzione, HealthOmics non memorizza nella cache gli output dell'attività.

> **Note**
>
> Se più attività in un'esecuzione utilizzano la stessa immagine del contenitore, HealthOmics fornisce la stessa versione dell'immagine a tutte queste attività. Dopo aver HealthOmics

estratto l'immagine, ignora eventuali aggiornamenti all'immagine del contenitore per tutta la durata dell'esecuzione. Questo approccio offre un'esperienza prevedibile e coerente e previene i potenziali problemi che potrebbero derivare dagli aggiornamenti dell'immagine del contenitore distribuiti durante l'esecuzione.

Esegui le prestazioni della cache

Quando attivi la memorizzazione nella cache delle chiamate per una corsa, potresti notare i seguenti impatti sulle prestazioni di esecuzione:

- Durante la prima esecuzione, HealthOmics salva i dati della cache per le attività in esecuzione. È possibile che si verifichino tempi di esportazione più lunghi per questa esecuzione, poiché la memorizzazione nella cache delle chiamate aumenta la quantità di dati di esportazione.
- Nelle esecuzioni successive, quando si riprende un'esecuzione dalla cache, è possibile ridurre il numero di fasi di elaborazione e ridurre il tempo di esecuzione.
- Se scegli di dichiarare anche i file intermedi come output, i tempi di esportazione potrebbero essere ancora più lunghi, poiché questi dati possono essere più dettagliati.

Eventi di conservazione e invalidazione dei dati della cache

Lo scopo principale di una cache di esecuzione è ottimizzare il calcolo delle attività in esecuzione. Se esiste una voce di cache valida corrispondente per un'attività, HealthOmics utilizza la voce della cache anziché ricalcolare l'attività. In caso contrario, HealthOmics torna al comportamento predefinito del servizio, che consiste nel ricalcolare l'attività e le attività da essa dipendenti. Utilizzando questo approccio, gli errori nella cache non causano il fallimento dell'esecuzione.

Ti consigliamo di gestire la dimensione della cache di esecuzione. Nel tempo, le voci della cache potrebbero non essere più valide a causa degli aggiornamenti del motore del flusso di lavoro o del HealthOmics servizio o a causa delle modifiche apportate durante l'esecuzione o le attività di esecuzione. Le seguenti sezioni forniscono ulteriori dettagli.

Argomenti

Aggiornamenti della versione manifesto e aggiornamento dei dati

Periodicamente, il HealthOmics servizio può introdurre nuove funzionalità o aggiornamenti del motore di flusso di lavoro che invalidano alcune o tutte le voci della cache di esecuzione. In questa situazione, è possibile che si verifichi un'unica perdita della cache nelle esecuzioni.

HealthOmics crea un file manifest JSON per ogni voce della cache. Per le esecuzioni iniziate dopo il 12 febbraio 2025, il file manifest include un parametro di versione. Se un aggiornamento del servizio invalida qualsiasi voce della cache, HealthOmics incrementa il numero di versione in modo da poter identificare le voci della cache legacy da rimuovere.

L'esempio seguente mostra un file manifesto con la versione impostata su 2:

```
{
     "arn": "arn:aws:omics:us-west-2:12345678901:runCache/0123456/
cacheEntry/1234567-195f-3921-a1fa-ffffcef0a6a4",
     "s3uri": "s3://example/1234567-d0d1-e230-
d599-10f1539f4a32/1348677/4795326/7e8c69b1-145f-3991-a1fa-ffffcef0a6a4",
     "taskArn": "arn:aws:omics:us-west-2:12345678901:task/4567891",
     "workDir": "/mnt/workflow/1234567-d0d1-e230-d599-10f1539f4a32/workdir/call-
TxtFileCopyTask/5w6tn5feyga7noasjuecdeoqpkltrfo3/wxz2fuddlo6hc4uh5s2lreaayczduxdm",
     "files": [
         {
             "name": "output_txt_file",
             "path": "out/output_txt_file/outfile.txt",
             "etag": "ajdhyg9736b9654673b9fbb486753bc8"
         }
     ],
     "nextflowContext": {},
     "otherOutputs": {},
     "version": 2,
  }
```

Per le esecuzioni con voci della cache che non sono più valide, ricostruisci la cache per creare nuove voci valide. Eseguite i seguenti passaggi per ogni esecuzione:

1. Avvia l'esecuzione una sola volta con la conservazione della cache impostata su CACHE ALWAYS. Questa esecuzione crea le nuove voci della cache.
2. Per le esecuzioni successive, imposta la conservazione della cache sull'impostazione precedente (CACHE ALWAYS o CACHE ON FAILURE).

Per pulire le voci della cache che non sono più valide, puoi eliminare queste voci dalla cache del bucket Amazon S3. HealthOmics non riutilizza mai queste voci della cache. Se scegli di conservare le voci non valide, non vi è alcun impatto sulle tue esecuzioni.

> **Note**
>
> La memorizzazione nella cache delle chiamate salva i dati di output delle attività nella posizione Amazon S3 specificata per la cache, il che comporta costi a carico dell'utente. Account AWS

Comportamento della cache di esecuzione

È possibile impostare il comportamento di esecuzione della cache per salvare gli output delle attività per le esecuzioni che non riescono (cache in caso di errore) o per tutte le esecuzioni (cache sempre). Quando si crea una cache di esecuzione, si imposta il comportamento predefinito della cache per tutte le esecuzioni che utilizzano tale cache. È possibile sovrascrivere il comportamento predefinito quando si avvia un'esecuzione.

Cache on failureè utile se si esegue il debug di un flusso di lavoro che non riesce dopo che diverse attività sono state completate correttamente. L'esecuzione successiva riprende dall'ultima attività completata con successo se tutte le variabili univoche considerate dall'hash sono identiche all'esecuzione precedente.

Cache alwaysè utile se si aggiorna un'attività in un flusso di lavoro completato correttamente. Ti consigliamo di seguire questi passaggi:

1. Crea una nuova corsa. Imposta sempre il comportamento della cache su Cache e avvia l'esecuzione.
2. Al termine dell'esecuzione, aggiorna l'attività nel flusso di lavoro e avvia una nuova esecuzione con il comportamento impostato su Cache always. Questa esecuzione elabora l'attività aggiornata e tutte le attività successive che dipendono dall'attività aggiornata. Tutte le altre attività utilizzano i risultati memorizzati nella cache.
3. Ripetere il passaggio 2 come richiesto, fino al completamento dello sviluppo dell'attività aggiornata.
4. Usa l'attività aggiornata secondo necessità per le esecuzioni future. Ricorda di passare alla cache le esecuzioni successive in caso di errore se prevedi di utilizzare input nuovi o diversi per queste esecuzioni.

> **Note**
>
> Consigliamo la modalità Cache always quando si utilizza lo stesso set di dati di test, ma non per un batch di esecuzioni. Se imposti questa modalità per un grande batch di esecuzioni, il sistema può esportare grandi quantità di dati su Amazon S3, con conseguente aumento dei tempi di esportazione e dei costi di storage.

Controlla la dimensione della cache di esecuzione

HealthOmics non elimina o archivia automaticamente i dati di esecuzione della cache né applica le regole di pulizia di Amazon S3 per la gestione dei dati della cache. Ti consigliamo di eseguire regolarmente la pulizia della cache per risparmiare sui costi di storage di Amazon S3 e per mantenere gestibili le dimensioni della cache di esecuzione. Puoi eliminare i file direttamente o impostare retention/replication politiche relative ai dati nel bucket run cache.

Ad esempio, puoi configurare una policy del ciclo di vita di Amazon S3 per far scadere gli oggetti dopo 90 giorni oppure puoi pulire manualmente i dati della cache alla fine di ogni progetto di sviluppo.

Le seguenti informazioni possono aiutarti a gestire le dimensioni dei dati della cache:

- Puoi visualizzare la quantità di dati presenti nella cache controllando Amazon S3. HealthOmics non monitora né segnala le dimensioni della cache.
- Se si elimina una voce di cache valida, l'esecuzione successiva non ha esito negativo. HealthOmics ricalcola l'attività e le relative attività dipendenti.
- Se modificate i nomi della cache o le strutture di directory in modo tale da non HealthOmics trovare una voce corrispondente per un'attività, HealthOmics ricalcola l'attività.

Se devi verificare se una voce della cache è ancora valida, controlla il numero di versione del manifesto della cache. Per ulteriori informazioni, consulta Aggiornamenti della versione manifesto e aggiornamento dei dati.

Creazione di una cache di esecuzione

Quando crei una cache di esecuzione, specifichi una posizione Amazon S3 per i dati della cache. Questi dati devono essere immediatamente accessibili. Il caching delle chiamate non recupera gli oggetti archiviati in Glacier (come le classi di archiviazione GFR e GDA).

Se il bucket Amazon S3 per i dati della cache è di proprietà di un altro Account AWS, fornisci quell'ID account quando crei la cache di esecuzione.

Creazione di una cache di esecuzione utilizzando la console

Dalla console, segui questi passaggi per creare una cache di esecuzione.

1. Apri la HealthOmics console.
2. Se necessario, apri il pannello di navigazione a sinistra (≡). Scegli Esegui cache.
3. Dalla pagina Esegui cache, scegli Crea cache di esecuzione.
4. Nel pannello dei dettagli della cache di esecuzione della pagina Crea cache di esecuzione, configura questi campi:
   a. Immettete un nome per la cache di esecuzione.
   b. (Opzionale) Immettere una descrizione.
   c. Inserisci una posizione S3 per l'output memorizzato nella cache. Scegli un bucket nella stessa regione del tuo flusso di lavoro.
   d. (Facoltativo) Inserisci il nome Account AWS del proprietario del bucket per verificare la proprietà del bucket. Se non inserisci un valore, il valore predefinito è l'ID del tuo account.
   e. In Comportamento della cache, configura il comportamento predefinito (se memorizzare nella cache gli output per le esecuzioni non riuscite o per tutte le esecuzioni). Quando avvii un'esecuzione, puoi opzionalmente sovrascrivere il comportamento predefinito.
5. (Facoltativo) Associate uno o più tag alla cache di esecuzione.
6. Scegli Crea cache di esecuzione. La console visualizza la nuova cache di esecuzione nella tabella Run caches.

Creazione di una cache di esecuzione utilizzando la CLI

Usa il comando create-run-cacheCLI per creare una cache di esecuzione. Il comportamento predefinito della cache è`CACHE_ON_FAILURE`.

```
aws omics create-run-cache \
      --name "workflow 123 run cache" \
      --description "my run cache" \
      --cache-s3-location "s3://amzn-s3-demo-bucket" \
      --cache-behavior "CACHE_ALWAYS"                \
```

```
      --cache-bucket-owner-id  "111122223333"
```

Se la creazione ha esito positivo, riceverai una risposta con i seguenti campi.

```
{
  "arn": "string",
  "id": "string",
  "status": "ACTIVE"
  "tags": {}
  }
```

Aggiornamento di una cache di esecuzione

Puoi modificare il nome, la descrizione, i tag o il comportamento della cache, ma non la posizione S3 della cache.

Aggiornamento di una cache di esecuzione tramite la console

Dalla console, segui questi passaggi per aggiornare una cache di esecuzione.

1. Apri la HealthOmics console.
2. Se necessario, apri il pannello di navigazione a sinistra (≡). Scegli Esegui cache.
3. Nella tabella Esegui cache, scegli la cache di esecuzione da aggiornare, quindi scegli Modifica.
4. Nel pannello dei dettagli di Run cache, puoi aggiornare i campi del nome, della descrizione e del comportamento della cache di esecuzione.
5. (Facoltativo) Associa uno o più nuovi tag alla cache di esecuzione o rimuovi i tag esistenti.
6. Scegliete Salva cache di esecuzione.

Aggiornamento di una cache di esecuzione utilizzando la CLI

Usa il comando update-run-cacheCLI per aggiornare una cache di esecuzione.

```
aws omics update-run-cache \
      --name "workflow 123 run cache" \
      --id "workflow id" \
      --description "my run cache" \
```

```
    --cache-behavior "CACHE_ALWAYS"
```

Se l'aggiornamento ha esito positivo, riceverai una risposta senza campi di dati.

Eliminazione di una cache di esecuzione

Puoi eliminare una cache di esecuzione se nessuna esecuzione attiva la utilizza. Se alcune esecuzioni utilizzano la cache di esecuzione, attendi il completamento delle esecuzioni oppure puoi annullare le esecuzioni.

L'eliminazione di una cache di esecuzione rimuove la risorsa e i relativi metadati, ma non elimina i dati in Amazon S3. Dopo aver eliminato la cache, non puoi ricollegarla o utilizzarla per le esecuzioni successive.

I dati memorizzati nella cache rimangono in Amazon S3 per l'ispezione. Puoi rimuovere i vecchi dati della cache utilizzando le operazioni standard di Delete S3. In alternativa, crea una policy sul ciclo di vita di Amazon S3 per far scadere i dati memorizzati nella cache che non usi più.

Eliminazione di una cache di esecuzione tramite la console

Dalla console, segui questi passaggi per eliminare una cache di esecuzione.

1. Apri la HealthOmics console.
2. Se necessario, apri il pannello di navigazione a sinistra (≡). Scegli Esegui cache.
3. Dalla tabella Run caches, scegli la cache di esecuzione da eliminare.
4. Dal menu della tabella Esegui cache, scegli Elimina.
5. Dalla finestra di dialogo modale, salva il link dati della cache Amazon S3 per riferimenti futuri, quindi conferma che desideri eliminare la cache di esecuzione.

   Puoi utilizzare il link Amazon S3 per ispezionare i dati memorizzati nella cache, ma non puoi ricollegare i dati a un'altra cache di esecuzione. Elimina i dati della cache al termine dell'ispezione.

Eliminazione di una cache di esecuzione utilizzando la CLI

Usa il comando delete-run-cacheCLI per eliminare una cache di esecuzione.

```
aws omics delete-run-cache \
```

```
    --id "my cache id"
```

Se l'eliminazione ha esito positivo, riceverai una risposta senza campi di dati.

Contenuto di una cache di esecuzione

HealthOmics organizza la cache di esecuzione con la seguente struttura nel bucket S3:

```
s3://{cache.S3location}/{cache.uuid}/runID/taskID/{cacheentry.uuid}/
```

Il cache.uuid è l'id univoco globale per la cache. Il cacheentry.uuid è l'uuid unico a livello globale per un'operazione memorizzata nella cache. HealthOmics assegna gli uuid alle cache e alle attività.

Per tutti i motori di workflow, la cache contiene i seguenti file:

- Il {cacheentryuuid}.json file: HealthOmics crea questo file manifesto, che contiene informazioni sulla cache, incluso un elenco di tutti gli elementi presenti nella cache e la versione della cache.
- File di output delle attività: l'output di ogni attività è costituito da uno o più file, come definito dall'attività.

Per un flusso di lavoro che utilizza Nextflow, il motore Nextflow crea questi file aggiuntivi nella cache:

- Il command.out file: questo file contiene il contenuto dello stdout per l'esecuzione dell'attività.
- Il .exitcode file: questo file contiene il codice di uscita dell'attività (un numero intero).

> **Note**
>
> Se desideri accedere ai file di attività intermedi nella cache di esecuzione per una risoluzione avanzata dei problemi, dichiara questi file come output delle attività nella definizione del flusso di lavoro.

Funzionalità di memorizzazione nella cache specifiche del motore

HealthOmics cerca di fornire un'implementazione coerente della memorizzazione nella cache delle chiamate nei motori di flusso di lavoro. Esistono alcune differenze in base al modo in cui ogni motore di workflow gestisce casi specifici:

- Nextflow
  - La memorizzazione nella cache tra diverse versioni di Nextflow non è garantita. Ad esempio, se esegui un'attività nella v23.10.0 e successivamente esegui la stessa attività nella v24.10.8, HealthOmics potresti considerare la seconda esecuzione come un errore nella cache.
  - È possibile disattivare la memorizzazione nella cache per singole attività utilizzando la direttiva cache. false Per informazioni su questa direttiva, consulta la specifica Processi nella specifica Nextflow.
  - HealthOmics utilizza la modalità clemente di Nextflow, ma non supporta la modalità deep caching.
  - La memorizzazione nella cache valuta ogni singolo oggetto S3 se si utilizza un pattern a glob nel percorso S3 degli input per un'attività. Se aggiungi un nuovo oggetto, HealthOmics ricalcola solo le attività che utilizzano il nuovo oggetto.
  - HealthOmics non memorizza nella cache i nuovi tentativi di attività. Questo comportamento è coerente con il comportamento predefinito di Nextflow.
- WDL
  - HealthOmics supporta il nuovo tipo di «directory» per gli input quando si utilizza la versione di sviluppo del flusso di lavoro WDL. Per la memorizzazione nella cache delle chiamate, se un oggetto nella directory cambia, HealthOmics ricalcola tutte le attività che entrano nella directory.
  - HealthOmics supporta la memorizzazione nella cache a livello di attività, ma non la memorizzazione nella cache a livello di flusso di lavoro.
  - È possibile disabilitare la memorizzazione nella cache per singole attività utilizzando l'attributo volatile. Per ulteriori informazioni, consulta Disattiva la memorizzazione nella cache a livello di attività con l'attributo volatile.
- CWL
  - I risultati costanti delle attività non sono esplicitamente visibili nei manifesti. HealthOmics memorizza nella cache gli output costanti come file intermedi.
  - È possibile controllare la memorizzazione nella cache per singole attività utilizzando la funzione. WorkReuse

Utilizzo della cache di esecuzione

Per impostazione predefinita, le esecuzioni non utilizzano una cache di esecuzione. Per utilizzare una cache per l'esecuzione, è necessario specificare la cache di esecuzione e il comportamento della cache di esecuzione all'avvio dell'esecuzione.

Al termine di un'esecuzione, puoi utilizzare la console, CloudWatch i registri o le operazioni API per tenere traccia degli accessi alla cache o risolvere i problemi relativi alla cache. Per informazioni dettagliate, consulta Monitoraggio delle informazioni relative alla memorizzazione delle chiamate e Risoluzione dei problemi relativi alla memorizzazione nella cache delle chiamate.

Se una o più attività in un'esecuzione generano output non deterministici, ti consigliamo vivamente di non utilizzare la memorizzazione nella cache delle chiamate per l'esecuzione o di disattivare queste attività specifiche dalla memorizzazione nella cache. Per ulteriori informazioni, consulta Modello di responsabilità condivisa.

> **Note**
>
> Fornisci un ruolo di servizio IAM quando avvii un'esecuzione. Per utilizzare la memorizzazione nella cache delle chiamate, il ruolo di servizio necessita dell'autorizzazione per accedere alla posizione di esecuzione della cache di Amazon S3. Per ulteriori informazioni, consulta Ruoli di servizio per AWS HealthOmics.

Puoi utilizzare Amazon Q CLI per analizzare e gestire i dati della cache di esecuzione. Per ulteriori informazioni, consulta le istruzioni di esempio per la CLI di Amazon Q e il tutorial sull'intelligenza artificiale generativa HealthOmics Agentic su. GitHub

Argomenti

- Configurazione di un run with run cache utilizzando la console
- Configurazione di un'esecuzione con cache di esecuzione utilizzando la CLI
- Casi di errore per le cache di esecuzione
- Monitoraggio delle informazioni relative alla memorizzazione delle chiamate

Configurazione di un run with run cache utilizzando la console

Dalla console, si configura la cache di esecuzione per un'esecuzione all'avvio dell'esecuzione.

1. Apri la HealthOmics console.
2. Se necessario, apri il pannello di navigazione a sinistra (≡). Scegli Runs.
3. Nella pagina Esecuzioni, scegli la corsa da iniziare.

4. Scegli Avvia esecuzione e completa i passaggi 1 e 2 di Avvia corsa come descritto inAvvio di una corsa utilizzando la console.
5. Nel passaggio 3 di Avvia esecuzione, scegli Seleziona una cache di esecuzione esistente.
6. Seleziona la cache dall'elenco a discesa Run cache ID.
7. Per ignorare il comportamento predefinito di esecuzione della cache, scegli il comportamento della cache per l'esecuzione. Per ulteriori informazioni, consulta Comportamento della cache di esecuzione.
8. Continuate con il passaggio 4 di Start run.

Configurazione di un'esecuzione con cache di esecuzione utilizzando la CLI

Per avviare un'esecuzione che utilizza una cache di esecuzione, aggiungi il parametro cache-id al comando CLI start-run. Facoltativamente, utilizzate il `cache-behavior` parametro per sovrascrivere il comportamento predefinito configurato per la cache di esecuzione. L'esempio seguente mostra solo i campi della cache per il comando:

```
aws omics start-run \
        ...
      --cache-id "xxxxxx"    \
      --cache-behavior  CACHE_ALWAYS
```

Se l'operazione ha esito positivo, si riceve una risposta senza campi di dati.

Casi di errore per le cache di esecuzione

Nei seguenti scenari, non è HealthOmics possibile memorizzare nella cache gli output delle attività, anche nel caso di un'esecuzione con il comportamento della cache impostato su Cache always.

- Se l'esecuzione rileva un errore prima che la prima attività venga completata correttamente, non ci sono output della cache da esportare.
- Se il processo di esportazione fallisce, HealthOmics non salva gli output delle attività nella posizione cache di Amazon S3.
- Se l'esecuzione fallisce a causa di un filesystem out of space errore, la memorizzazione nella cache delle chiamate non salva alcun output dell'attività.
- Se si annulla un'esecuzione, la memorizzazione nella cache delle chiamate non salva alcun output dell'attività.

- Se l'esecuzione presenta un timeout di esecuzione, la memorizzazione nella cache delle chiamate non salva alcun output dell'attività, anche se l'esecuzione è stata configurata per utilizzare la cache in caso di errore.

Monitoraggio delle informazioni relative alla memorizzazione delle chiamate

È possibile tenere traccia degli eventi di memorizzazione nella cache delle chiamate (come gli accessi alla cache di esecuzione) utilizzando la console, la CLI o i registri. CloudWatch

Argomenti

Tieni traccia degli accessi alla cache utilizzando la console

Nella pagina dei dettagli di esecuzione di un'esecuzione, la tabella Esegui attività visualizza le informazioni sugli accessi alla cache per ogni attività. La tabella include anche un collegamento alla voce della cache associata. Utilizzare la procedura seguente per visualizzare le informazioni relative agli accessi alla cache durante un'esecuzione.

1. Apri la HealthOmics console.
2. Se necessario, aprite il pannello di navigazione a sinistra (≡). Scegli Runs.
3. Nella pagina Esecuzioni, scegli la corsa da ispezionare.
4. Nella pagina dei dettagli dell'esecuzione, scegli la scheda Esegui attività per visualizzare la tabella delle attività.
5. Se un'attività ha un accesso alla cache, la colonna Cache hit contiene un collegamento alla posizione di accesso alla cache di esecuzione in Amazon S3.
6. Scegli il link per controllare la voce di run cache.

Tieni traccia della memorizzazione nella cache delle chiamate utilizzando la CLI

Utilizza il comando CLI get-run per confermare se l'esecuzione ha utilizzato una cache delle chiamate.

```
aws omics get-run --id 1234567
```

Nella risposta, se il `cacheId` campo è impostato, l'esecuzione utilizza quella cache.

Utilizza il comando list-run-tasksCLI per recuperare la posizione dei dati della cache per ogni attività memorizzata nella cache in esecuzione.

```
aws omics list-run-tasks --id 1234567
```

Nella risposta, se il campo CacheHit per un'attività è vero, il campo Caches3URI fornisce la posizione dei dati della cache per quell'attività.

Puoi anche utilizzare il comando get-run-taskCLI per recuperare la posizione dei dati della cache per un'attività specifica:

```
aws omics get-run-task --id 1234567 --task-id <task_id>
```

Tieni traccia della memorizzazione nella cache delle chiamate utilizzando Logs CloudWatch

HealthOmics crea registri delle attività della cache nel gruppo di registri. `/aws/omics/WorkflowLog` CloudWatch <cache_id><cache_uuid>Esiste un flusso di log per ogni cache di esecuzione: runCache//.

Per le esecuzioni che utilizzano la memorizzazione nella cache delle chiamate, HealthOmics genera voci di CloudWatch registro per questi eventi:

- creazione di una voce nella cache (CACHE_ENTRY_CREATED)
- corrispondente a una voce della cache (CACHE_HIT)
- non riesce a far corrispondere una voce della cache (CACHE_MISS)

Per ulteriori informazioni su questi registri, vedere. Effettua il login CloudWatch

Utilizza la seguente query CloudWatch Insights sul gruppo di `/aws/omics/WorkflowLog` log per restituire il numero di accessi alla cache per esecuzione per questa cache:

```
filter @logStream like 'runCache/<CACHE_ID>/'
 fields @timestamp, @message
 filter logMessage like 'CACHE_HIT'
 parse "run: *," as run
 stats count(*) as cacheHits by run
```

Utilizzate la seguente query per restituire il numero di voci della cache create da ogni esecuzione:

```
filter @logStream like 'runCache/<CACHE_ID>/'
 fields @timestamp, @message
 filter logMessage like 'CACHE_ENTRY_CREATED'
 parse "run: *," as run
 stats count(*) as cacheEntries by run
```

Condivisione dei flussi HealthOmics di lavoro

In qualità di proprietario di un flusso di lavoro privato, puoi condividere il flusso di lavoro con un Account AWS utente della stessa area. Per condividere un flusso di lavoro con più di uno Account AWS, devi creare più condivisioni dello stesso flusso di lavoro.

In qualità di proprietario, puoi revocare l'accesso a un flusso di lavoro condiviso eliminando la condivisione.

> **Note**
>
> HealthOmics consente automaticamente a un flusso di lavoro condiviso di accedere al repository Amazon ECR mentre il flusso di lavoro è in esecuzione nell'account dell'abbonato. Non è necessario concedere un accesso aggiuntivo al repository per i flussi di lavoro condivisi.

Quando condividi un flusso di lavoro, il sottoscrittore può utilizzare qualsiasi versione del flusso di lavoro. Se hai bisogno del controllo degli accessi a livello di versione per un flusso di lavoro condiviso, ti consigliamo di creare flussi di lavoro separati anziché utilizzare versioni del flusso di lavoro.

Argomenti

Sottoscrizione a un flusso di lavoro condiviso

Per iscriverti a un flusso di lavoro condiviso, segui questi passaggi generali per accettare e utilizzare il flusso di lavoro:

1. Usa la console o l'API per accettare la condivisione. Imposta la regione corrente sulla stessa regione della richiesta di condivisione.
   - Per trovare la richiesta di condivisione nella console, vai alla pagina Tutte le condivisioni di risorse, quindi scegli la scheda Condivisa con me.
2. Usa la console o l'API per creare un'esecuzione per il flusso di lavoro condiviso.
   - Per trovare la pagina dei dettagli del flusso di lavoro nella console, vai a Condiviso con me (vedi passaggio 1), quindi scegli il link Risorsa per il flusso di lavoro condiviso.
3. Fornisci i tuoi dati di input per il flusso di lavoro.
4. Il flusso di lavoro condiviso viene eseguito nel tuo Account AWS.

In qualità di abbonato a un flusso di lavoro condiviso, il sistema ti impedisce di eseguire le seguenti azioni del flusso di lavoro:

- Esportazione di un flusso di lavoro condiviso
- Riesecuzione del flusso di lavoro condiviso
  - Si crea una nuova esecuzione per il flusso di lavoro condiviso.
- Ricondivisione del flusso di lavoro.
- Assegnazione di un tag al flusso di lavoro.
- Eliminazione del flusso di lavoro.
  - Quando non è più necessario il flusso di lavoro, si elimina la condivisione del flusso di lavoro.

Condivisione di risorse tra account in AWS HealthOmicsPer ulteriori informazioni sulla condivisione delle risorse, vedere.

Monitoraggio dello stato di una condivisione del flusso di lavoro

HealthOmics invia un evento a EventBridge per ogni modifica di stato di una condivisione del flusso di lavoro. Se desideri ricevere notifiche su modifiche di stato specifiche, imposta una EventBridge regola per monitorare gli eventi di modifica dello stato di Workflow share. Ad esempio:

- Desideri ricevere una notifica ogni volta che ricevi una richiesta di condivisione del flusso di lavoro e ogni volta che un utente revoca una condivisione del flusso di lavoro.
- Dopo aver avviato una richiesta di condivisione del flusso di lavoro, desideri ricevere una notifica quando l'utente accetta o rifiuta la richiesta.

Per informazioni dettagliate sull'utilizzo degli eventi, consulta. Utilizzo EventBridge con AWS HealthOmics

Condivisione di un flusso di lavoro privato tramite la console

Dalla console, puoi condividere un flusso di lavoro privato con un Account AWS utente della stessa area del flusso di lavoro.

Per condividere un flusso di lavoro privato

1. Apri la HealthOmics console.
2. Se necessario, apri il riquadro di navigazione a sinistra (≡). Scegli Flussi di lavoro privati.
3. Dalla tabella Flussi di lavoro della pagina Flussi di lavoro privati, seleziona il flusso di lavoro da condividere e scegli Condividi.
4. Nel pannello Condividi dettagli della pagina Condividi flusso di lavoro, inserisci un nome descrittivo per la condivisione e inserisci il nome Account AWS del sottoscrittore.
5. Scegli Condividi risorsa. La console visualizza le condivisioni di risorse nella pagina Tutte le condivisioni di risorse.

Lo stato iniziale della condivisione è in sospeso. Dopo che il sottoscrittore ha accettato la condivisione, lo stato diventa attivo.

Condivisione di un flusso di lavoro privato tramite la CLI

Utilizza l'operazione API create-share per creare una condivisione del flusso di lavoro. Il sottoscrittore principale è Account AWS l'utente che avrà accesso al flusso di lavoro.

```
aws omics create-share \
  --resource-arn "arn:aws:omics:us-west-2:555555555555:workflow/123456" \
  --principal-subscriber "123456789012" \
  --name "my_Share-123"
```

Se la creazione ha esito positivo, riceverai una risposta con l'ID e lo stato della condivisione.

```
{
"shareId": "495c21bedc889d07d0ab69d710a6841e-dd75ab7a1a9c384fa848b5bd8e5a7e0a",
"name": "my_Share-123",
"status": "PENDING"
}
```

La condivisione rimane in sospeso fino a quando il sottoscrittore non la accetta tramite l'operazione `accept-share` API.

Vedi altri Condivisione di risorse tra account in AWS HealthOmics esempi di utilizzo delle API.

Accettazione di un flusso di lavoro condiviso tramite la console

È possibile utilizzare la console per accettare una condivisione del flusso di lavoro offerta. Assicurati di impostare la console nella stessa regione del flusso di lavoro.

1. Apri la HealthOmics console.
2. Se necessario, apri il pannello di navigazione a sinistra (≡). Scegli Tutte le condivisioni di risorse, quindi scegli la scheda Condivisi con me.
3. Dalla tabella Risorse condivise con me, seleziona la condivisione del flusso di lavoro, quindi scegli Accetta.

Dopo aver accettato il flusso di lavoro, scegli il link Risorsa per il flusso di lavoro condiviso per visualizzarne i dettagli.

Esecuzione di un flusso di lavoro condiviso tramite la console

Dopo aver accettato una condivisione del flusso di lavoro, puoi avviare un'esecuzione sul flusso di lavoro.

1. Apri la HealthOmics console.
2. Se necessario, apri il riquadro di navigazione a sinistra (≡). Scegli Tutte le condivisioni di risorse, quindi scegli la scheda Condivisi con me.
3. Dalla tabella Risorse condivise con me, scegli il link Risorse per il flusso di lavoro condiviso.
4. Nella pagina dei dettagli del flusso di lavoro, scegli Crea esegui.

La console apre la pagina Crea esecuzione, con il tipo di flusso di lavoro (condiviso) e l'ID del flusso di lavoro precompilati.

5. Configura i campi rimanenti nel modulo Create run. Per ulteriori informazioni, consulta Avvio di una corsa utilizzando la console.

Esecuzione di un flusso di lavoro condiviso utilizzando l'API

Usa get-workflow per recuperare l'ARN del flusso di lavoro condiviso.

```
aws omics get-workflow --id 1234567 \
--workflow-owner-id 55555555555
```

Quando esegui il flusso di lavoro, fornisci l' Account AWS ID del proprietario del flusso di lavoro e l'ARN del flusso di lavoro condiviso.

```
aws omics start-run --id 1234567 --workflow-owner-id 55555555555 \
--role-arn arn:aws:iam::1234567892012:role/service-role/OmicsWorkflow-20221004T164236 \
--name ArchiveTest --retention-mode REMOVE
```

Flussi di lavoro Ready2Run in HealthOmics

I flussi di lavoro Ready2Run sono flussi di lavoro preconfigurati pubblicati da editori di terze parti. Alcuni editori, come Sentieon Inc, offrono flussi di lavoro basati su abbonamento. Altri flussi di lavoro Ready2Run non richiedono un abbonamento e alcuni flussi di lavoro sono open source, come i flussi di lavoro NF-Core.

I flussi di lavoro Ready2Run si adattano bene ai seguenti scenari:

- Desiderate concentrarvi sull'analisi dell'output della pipeline e sulla generazione dei risultati, senza la necessità di configurare l'infrastruttura sottostante.
- Desiderate replicare i risultati utilizzando flussi di lavoro consolidati.
- In qualità di sviluppatore di software, desideri integrare la tua applicazione direttamente con l' HealthOmics SDK.

HealthOmics supporta il controllo delle versioni per i flussi di lavoro Ready2Run. Per un flusso di lavoro Ready2Run che offre versioni, è possibile specificare il nome della versione quando si avvia un'esecuzione.

Tutti i flussi di lavoro Ready2Run forniscono registri, compresi i log, che è possibile utilizzare per la risoluzione dei CloudWatch problemi.

> **Note**
>
> I flussi di lavoro Sentieon Ready2Run sono basati su abbonamento. Quando esegui un flusso di lavoro Sentieon Ready2Run per la prima volta in un account, Sentieon crea automaticamente una licenza di valutazione di due settimane per il tuo. Account AWS La licenza è valida per tutti i flussi di lavoro Sentieon Ready2Run. Al termine del periodo di valutazione, è possibile richiedere una licenza permanente o richiedere un'estensione della licenza di valutazione. Per informazioni dettagliate, vedi Subscribing to Sentieon Ready2Run workflows.

Argomenti

- Avvio dei flussi di HealthOmics lavoro Ready2Run tramite la console
- Avvio dei flussi di HealthOmics lavoro Ready2Run utilizzando l'API

Flussi di lavoro Ready2Run disponibili in HealthOmics

La tabella seguente elenca i flussi di lavoro Ready2Run disponibili in. HealthOmics

È possibile accedere alla HealthOmicsconsole per visualizzare informazioni dettagliate su questi flussi di lavoro, inclusi i parametri di input e i diagrammi di flusso di lavoro. Per informazioni sui prezzi dei flussi di lavoro Ready2Run, vedi Prezzi. HealthOmics

> **Note**
>
> Ogni flusso di lavoro Ready2Run ha una dimensione massima del file di input. Queste dimensioni massime di file non sono regolabili.

Nome del flusso di lavoro	Publisher	È richiesto un abbonamento?	Dimensione massima del file di input (GiB)	Tempo di esecuzione stimato (HH:MM)
AlphaFold per 601-1200 residui	Google DeepMind	No	1	11:15
AlphaFold per un massimo di 600 residui	Google DeepMind	No	1	7:30
Bases2Fastq per 2x150	Element Biosciences	No	1000	1:45
Bases2Fastq per 2x300	Element Biosciences	No	1000	1:30
Bases2Fastq per 2x75	Element Biosciences	No	500	0:45

Nome del flusso di lavoro	Publisher	È richiesto un abbonamento?	Dimensione massima del file di input (GiB)	Tempo di esecuzione stimato (HH:MM)
ESMFold per un massimo di 800 residui	Metaricerche	No	1	0:15
GATK-BP fq2bam	Broad Institute	No	64	10:10
GATK-BP Germline bam2vcf per genoma 30x	Broad Institute	No	39	2:45
GATK-BP Germline fq2vcf per genoma 30x	Broad Institute	No	64	12:30
GATK-BP WES somatico bam2vcf	Broad Institute	No	86	1:30
NVIDIA Parabricks BAM2 FQ2 BAM WGS per un massimo di 30 volte	NVIDIA Corporation	No	80	1:39
NVIDIA Parabricks BAM2 FQ2 BAM WGS per un massimo di 50 volte	NVIDIA Corporation	No	120	2:45

Nome del flusso di lavoro	Publisher	È richiesto un abbonamento?	Dimensione massima del file di input (GiB)	Tempo di esecuzione stimato (HH:MM)
NVIDIA Parabricks BAM2 FQ2 BAM WGS per un massimo di 5 volte	NVIDIA Corporation	No	20	0:18
NVIDIA Parabricks FQ2 BAM WGS per un massimo di 30 volte	NVIDIA Corporation	No	71	1:00
NVIDIA Parabricks FQ2 BAM WGS per un massimo di 50 volte	NVIDIA Corporation	No	137	1:45
NVIDIA Parabricks FQ2 BAM WGS per un massimo di 5 volte	NVIDIA Corporation	No	13	0:15
NVIDIA Parabricks DeepVariant Germline WGS per un massimo di 30 volte	NVIDIA Corporation	No	71	2:00

Nome del flusso di lavoro	Publisher	È richiesto un abbonamento?	Dimensione massima del file di input (GiB)	Tempo di esecuzione stimato (HH:MM)
NVIDIA Parabricks DeepVariant Germline WGS per un massimo di 50 volte	NVIDIA Corporation	No	137	3:30
NVIDIA Parabricks DeepVariant Germline WGS per un massimo di 5 volte	NVIDIA Corporation	No	12	0:30
NVIDIA Parabricks HaplotypeCaller Germline WGS per un massimo di 30 volte	NVIDIA Corporation	No	71	1:15
NVIDIA Parabricks HaplotypeCaller Germline WGS per un massimo di 50X	NVIDIA Corporation	No	137	2:00

Nome del flusso di lavoro	Publisher	È richiesto un abbonamento?	Dimensione massima del file di input (GiB)	Tempo di esecuzione stimato (HH:MM)
NVIDIA Parabricks HaplotypeCaller Germline WGS per un massimo di 5 volte	NVIDIA Corporation	No	13	0:15
NVIDIA Parabricks Somatic Mutect2 WGS per un massimo di 50 volte	NVIDIA Corporation	No	196	0:45
sc RNAseq con Kallisto BUStools	Nucleo NF	No	119	1:30
sc RNAseq con salmone Alevin-fry	Nucleo NF	No	119	2:30
sc RNAseq con STARsolo	NF-Core	No	119	2:30
Sentieon Germline BAM WES per un massimo di 300 volte	Sentieon, Inc.	Sì	9	1:00
Sentieon Germline BAM WGS per un massimo di 32x	Sentieon, Inc.	Sì	18	13:30.

Nome del flusso di lavoro	Publisher	È richiesto un abbonamento?	Dimensione massima del file di input (GiB)	Tempo di esecuzione stimato (HH:MM)
Sentieon Germline FASTQ WES per un massimo di 100 volte	Sentieon, Inc.	Sì	5	0:45
Sentieon Germline FASTQ WES per un massimo di 300 volte	Sentieon, Inc.	Sì	26	2:00
Sentieon Germline FASTQ WGS per un massimo di 32x	Sentieon, Inc.	Sì	51	3:30
Sentieon per ONT LongRead	Sentieon, Inc.	Sì	25	13:30.
Sentieon LongRead per PacBio HiFi	Sentieon, Inc.	Sì	58	4:00
Sentieon Somatic WES	Sentieon, Inc.	Sì	50	2:30
SGS somatico Sentieron	Sentieon, Inc.	Sì	113	4:30
Ultima Genomics DeepVariant per un massimo di 40 volte	Ultima Genomica	No	91	1:55

Quando si utilizza un flusso di lavoro Ready2Run, il flusso di lavoro è preconfigurato e non può essere modificato. A differenza dei flussi di lavoro privati, i flussi di lavoro Ready2Run non supportano quanto segue:

- Aumento della dimensione massima del file di input
- Modifica delle risorse di elaborazione o esecuzione dello storage
- Modifica della definizione o dei contenitori del flusso di lavoro
- Aggiungere esecuzioni a un gruppo di esecuzioni
- Condivisione del flusso di lavoro

Se l'editore ha condiviso il flusso di lavoro Ready2Run su GitHub, puoi creare il tuo flusso di lavoro privato basato sul flusso di lavoro Ready2Run. La tabella seguente fornisce collegamenti ai flussi di lavoro per ogni editore. GitHub

Publisher	Flussi di lavoro su GitHub
Google DeepMind, Meta Research	Flussi di lavoro per la ripiegatura
Element Biosciences	Per informazioni, contatta Element Biosciences
Broad Institute	Flussi di lavoro GATK
NVIDIA Corporation	Flussi di lavoro Parabricks
nf-core	Flussi di lavoro NF-Core
Sentiero	Flussi di lavoro Sentieon
Ultima Genomica	Flussi di lavoro di Ultima Genomics

Iscrizione ai flussi di lavoro Sentieon Ready2Run

I flussi di lavoro Sentieon Ready2Run sono basati su abbonamento. Quando esegui un flusso di lavoro Sentieon Ready2Run per la prima volta in un account, Sentieon crea automaticamente una licenza di valutazione di due settimane per il tuo. Account AWS La licenza è valida per tutti i flussi di lavoro Sentieon Ready2Run. Al termine del periodo di valutazione, è possibile richiedere una licenza permanente o richiedere un'estensione della licenza di valutazione.

Segui questi passaggi per iscriverti ai flussi di lavoro Sentieon Ready2Run:

- Trova il tuo ID utente AWS Canonical seguendo queste istruzioni.
- Invia un'e-mail al gruppo di supporto Sentieon (support@sentieon.com) per richiedere una licenza software. Fornisci il tuo ID utente AWS Canonical nell'e-mail.

Avvio dei flussi di HealthOmics lavoro Ready2Run tramite la console

L'utilizzo dei flussi di lavoro Ready2Run nella console è simile all'utilizzo di un flusso di lavoro privato. Una differenza fondamentale è che Workflow Publisher fornisce dati di esempio, in modo da poter provare il flusso di lavoro senza creare dati propri.

Per utilizzare un flusso di lavoro Ready2Run nella console

1. Apri la HealthOmics console.
2. Se necessario, apri il riquadro di navigazione a sinistra (≡). Scegli i flussi di lavoro Ready2Run.
3. Nella pagina Flussi di lavoro Ready2Run, scegli il flusso di lavoro che desideri utilizzare. La console apre la pagina dei dettagli per quel flusso di lavoro.
4. La scheda dei dettagli elenca informazioni come il nome, il prezzo di listino per esecuzione, la descrizione, il tipo di linguaggio del flusso di lavoro, la capacità di archiviazione dell'esecuzione, lo stato, la data di creazione e i parametri con descrizioni. La scheda dei dettagli indica anche se il flusso di lavoro richiede un abbonamento.
5. Per utilizzare il flusso di lavoro, scegli Crea ed esegui
6. Nella pagina Specificare i dettagli della corsa, inserisci un nome di esecuzione. Facoltativamente, è possibile specificare la versione del flusso di lavoro. È inoltre possibile aggiungere la priorità di esecuzione all'esecuzione.
7. Inserisci o seleziona una posizione Amazon S3 per l'output di esecuzione.
8. Per la modalità di conservazione dei metadati di Run, scegli se conservare o rimuovere i metadati di esecuzione.
9. Nel pannello Ruolo di servizio, scegli se utilizzare un ruolo di servizio esistente o crearne uno nuovo.
10. (Facoltativo) Aggiungi tag per identificare e gestire la tua corsa.
11. Scegli Next (Successivo).

12. Dalla pagina Aggiungi parametri, scegli una delle opzioni per aggiungere i valori dei parametri di esecuzione:
    - Seleziona un file di parametri (in formato JSON) da una posizione Amazon S3.
    - Seleziona un file di parametri (in formato JSON) dall'unità locale.
    - Immettete manualmente i valori dei parametri.
    - Esegui il flusso di lavoro con i dati di esempio di Ready2Run forniti dall'editore del flusso di lavoro.
13. Se carichi un file JSON, la console analizza il file ed esegue la convalida in linea. È quindi possibile aggiornare manualmente i valori dei parametri in base alle esigenze.
14. Scegli Next (Successivo).
15. Controlla i dati immessi, quindi scegli Avvia esecuzione.

Avvio dei flussi di HealthOmics lavoro Ready2Run utilizzando l'API

La maggior parte delle operazioni API si comporta in modo simile per i flussi di lavoro Ready2Run e per i flussi di lavoro privati.

Per restituire un elenco di flussi di lavoro Ready2Run disponibili, utilizza list-workflows con il parametro impostato su RUN. `type` READY2

```
aws omics list-workflows --type READY2RUN
```

Dopo aver identificato il flusso di lavoro da eseguire dalla risposta list-workflows, puoi utilizzare get-workflow con il parametro per ottenere maggiori dettagli. `--id`

```
aws omics get-workflow --type READY2RUN --id workflow id
```

Per eseguire un flusso di lavoro Ready2Run, è possibile utilizzare l'operazione API start-run con il parametro workflow-type impostato su, come illustrato nell'esempio seguente `READY2RUN`

```
aws-omics start-run \
  --workflow-type READY2RUN \
  --workflow-id workflow id \
  --output-uri &example-s3-bucket; \
  --role-arn arn:aws:iam::1234567892012:role/service-role/OmicsWorkflow-20221004T164236
 \
```

```
  --parameters file:///path/to/parameters.json
```

Per specificare una versione del flusso di lavoro, utilizzate il parametro workflow-version, come illustrato in questo esempio.

```
aws-omics start-run \
  --workflow-type READY2RUN \
  ...
  --version-name '3.0.0'
```

Per monitorare l'esecuzione, è possibile utilizzare l'operazione API get-run, come illustrato.

```
aws-omics get-run \
  --id run id
```

HealthOmics archiviazione

Utilizza HealthOmics lo storage per archiviare, recuperare, organizzare e condividere i dati genomici in modo efficiente e a basso costo. HealthOmics lo storage comprende le relazioni tra diversi oggetti di dati, in modo da poter definire quali set di lettura hanno avuto origine dalla stessa fonte di dati. Ciò fornisce la provenienza dei dati.

I dati archiviati nello `ACTIVE` stato sono recuperabili immediatamente. I dati a cui non si accede da 30 giorni o più vengono archiviati nello `ARCHIVE` stato. Per accedere ai dati archiviati, puoi riattivarli tramite le operazioni o la console dell'API.

HealthOmics gli archivi di sequenze sono progettati per preservare l'integrità del contenuto dei file. Tuttavia, l'equivalenza bit per bit dei file di dati importati e dei file esportati non viene preservata a causa della compressione durante il tiering attivo e di archiviazione.

Durante l'ingestione, HealthOmics genera un tag di entità o HealthOmics ETagconsente di convalidare l'integrità del contenuto dei file di dati. Le porzioni di sequenziamento vengono identificate e acquisite ETag a livello di origine di un set di lettura. Il ETag calcolo non altera il file effettivo o i dati genomici. Dopo la creazione di un set di lettura, non ETag dovrebbe cambiare durante il ciclo di vita della sorgente del set di lettura. Ciò significa che la reimportazione dello stesso file comporta il calcolo dello stesso ETag valore.

Argomenti

HealthOmics ETags e provenienza dei dati

Un HealthOmics ETag (tag di entità) è un hash del contenuto acquisito in un archivio di sequenze. Ciò semplifica il recupero e l'elaborazione dei dati mantenendo al contempo l'integrità dei contenuti dei file di dati acquisiti. ETag Riflette le modifiche al contenuto semantico dell'oggetto, non ai suoi metadati. Il tipo di set di lettura e l'algoritmo specificati determinano la modalità di calcolo ETag . Il ETag calcolo non altera il file effettivo o i dati genomici. Quando lo schema del tipo di file del set di lettura lo consente, l'archivio delle sequenze aggiorna i campi collegati alla provenienza dei dati.

I file hanno un'identità bit per bit e un'identità semantica. L'identità bit per bit significa che i bit di un file sono identici e un'identità semantica significa che i contenuti di un file sono identici. L'identità semantica è resistente alle modifiche dei metadati e alle modifiche di compressione poiché acquisisce l'integrità del contenuto del file.

I set di lettura negli archivi di HealthOmics sequenza sono sottoposti a compression/decompression cicli e al monitoraggio della provenienza dei dati durante tutto il ciclo di vita di un oggetto. Durante questa elaborazione, l'identità bit per bit di un file ingerito può cambiare e dovrebbe cambiare ogni volta che viene attivato un file; tuttavia, l'identità semantica del file viene mantenuta. L'identità semantica viene acquisita come tag di HealthOmics entità, oppure ETag viene calcolata durante l'inserimento del Sequence Store e disponibile come metadati del set di lettura.

Quando lo schema dei tipi di file del set di lettura lo consente, i campi degli aggiornamenti dell'archivio delle sequenze sono collegati alla provenienza dei dati. Per i file UBam, BAM e CRAM, viene aggiunto un nuovo `@CO` tag or all'intestazione. `Comment` Il commento contiene l'ID dell'archivio della sequenza e il timestamp di inserimento.

Amazon S3 ETags

Quando si accede a un file utilizzando l'URI di Amazon S3, le operazioni API di Amazon S3 possono anche restituire valori Amazon S3 e valori di checksum. ETag I valori di Amazon S3 ETag e checksum differiscono da quelli HealthOmics ETags perché rappresentano l'identità bit per bit del file. Per ulteriori informazioni sui metadati e sugli oggetti descrittivi, consulta la documentazione dell'API Amazon S3 Object. ETag I valori di Amazon S3 possono cambiare con ogni ciclo di attivazione di un set di lettura e puoi utilizzarli per convalidare la lettura di un file. Tuttavia, non memorizzare nella cache ETag i valori di Amazon S3 da utilizzare per la convalida dell'identità dei file durante il ciclo di vita del file perché non rimangono coerenti. Al contrario, HealthOmics ETag rimane coerente per tutto il ciclo di vita del set di lettura.

Come calcola HealthOmics ETags

ETag Viene generato da un hash del contenuto del file ingerito. La famiglia di ETag algoritmi è impostata come impostazione MD5up predefinita, ma può essere configurata in modo diverso durante la creazione dell'archivio di sequenze. Quando ETag viene calcolato, l'algoritmo e gli hash calcolati vengono aggiunti al set di lettura. MD5 Gli algoritmi supportati per i tipi di file sono i seguenti.

- FASTQ_ MD5up — Calcola l' MD5hash di una sorgente di lettura FASTQ completa e non compressa.
- BAM_ MD5up — Calcola l' MD5 hash della sezione di allineamento di una sorgente non compressa del set di lettura BAM o UBam rappresentata nel SAM, in base al riferimento collegato, se disponibile.
- CRAM_ MD5up — Calcola l' MD5 hash della sezione di allineamento della sorgente non compressa del set di lettura CRAM rappresentata nel SAM, in base al riferimento collegato.

> **Note**
>
> MD5 è noto che l'hashing è vulnerabile alle collisioni. Per questo motivo, due file diversi potrebbero avere le stesse caratteristiche ETag se fossero stati prodotti per sfruttare la collisione nota.

I seguenti algoritmi sono supportati per la famiglia. SHA256 Gli algoritmi vengono calcolati come segue:

- FASTQ_ SHA256up — Calcola l'hash SHA-256 di una sorgente di set di lettura FASTQ completa e non compressa.
- BAM_ SHA256up — Calcola l'hash SHA-256 della sezione di allineamento di una sorgente non compressa del set di lettura BAM o UBam rappresentata nel SAM, in base al riferimento collegato, se disponibile.
- CRAM_ SHA256up — Calcola l'hash SHA-256 della sezione di allineamento di una sorgente del set di lettura CRAM non compressa rappresentata nel SAM, in base al riferimento collegato.

I seguenti algoritmi sono supportati per la famiglia. SHA512 Gli algoritmi vengono calcolati come segue:

- FASTQ_ SHA512up — Calcola l'hash SHA-512 di una sorgente di set di lettura FASTQ completa e non compressa.
- BAM_ SHA512up — Calcola l'hash SHA-512 della sezione di allineamento di una sorgente non compressa del set di lettura BAM o UBam rappresentata nel SAM, in base al riferimento collegato, se disponibile.
- CRAM_ SHA512up — Calcola l'hash SHA-512 della sezione di allineamento di una sorgente del set di lettura CRAM non compressa rappresentata nel SAM, in base al riferimento collegato.

Creazione di un archivio di riferimento HealthOmics

Un archivio di riferimento in HealthOmics è un archivio dati per l'archiviazione dei genomi di riferimento. È possibile disporre di un unico archivio di riferimento in ciascuna Account AWS regione. È possibile creare un archivio di riferimento utilizzando la console o la CLI.

Argomenti

Creazione di un archivio di riferimento utilizzando la console

Come creare un archivio di riferimenti

1. Apri la HealthOmics console.
2. Se necessario, apri il riquadro di navigazione a sinistra (≡). Scegli Reference store.
3. Scegli Reference genomes tra le opzioni di archiviazione dei dati Genomics.
4. Puoi scegliere un genoma di riferimento importato in precedenza o importarne uno nuovo. Se non hai importato un genoma di riferimento, scegli Importa genoma di riferimento in alto a destra.
5. Nella pagina Crea processo di importazione del genoma di riferimento, scegli l'opzione Creazione rapida o Creazione manuale per creare un archivio di riferimento, quindi fornisci le seguenti informazioni.
   - Nome del genoma di riferimento: un nome univoco per questo negozio.
   - Descrizione (opzionale): una descrizione di questo archivio di riferimento.
   - Ruolo IAM: seleziona un ruolo con accesso al tuo genoma di riferimento.

- Riferimento da Amazon S3: seleziona il file di sequenza di riferimento in un bucket Amazon S3.
- Tag (opzionale): fornisci fino a 50 tag per questo negozio di riferimento.

Creazione di un archivio di riferimento utilizzando la CLI

L'esempio seguente mostra come creare un archivio di riferimento utilizzando. AWS CLIÈ possibile disporre di un archivio di riferimento per AWS regione.

Gli archivi di riferimento supportano l'archiviazione di file FASTA con le estensioni `.fasta.fa`,`.fas`,`.fsa`,`.faa`,`.fna`,`.ffn`,,`.frn`, `.mpfa.seq`,`.txt`. È supportata anche la `bgzip` versione di queste estensioni.

Nell'esempio seguente, sostituiscilo *`reference store name`* con il nome che hai scelto per il tuo negozio di riferimento.

```
aws omics create-reference-store --name "reference store name"
```

Riceverai una risposta JSON con l'ID e il nome dell'archivio di riferimento, l'ARN e il timestamp di quando è stato creato il tuo negozio di riferimento.

```
{
    "id": "3242349265",
    "arn": "arn:aws:omics:us-west-2:555555555555:referenceStore/3242349265",
    "name": "MyReferenceStore",
    "creationTime": "2022-07-01T20:58:42.878Z"
}
```

È possibile utilizzare l'ID dell'archivio di riferimento in comandi aggiuntivi. AWS CLI È possibile recuperare l'elenco degli store di riferimento IDs collegati al proprio account utilizzando il list-reference-storescomando, come illustrato nell'esempio seguente.

```
aws omics list-reference-stores
```

In risposta, riceverai il nome del tuo negozio di riferimento appena creato.

```
{
    "referenceStores": [
        {
```

```
            "id": "3242349265",
            "arn": "arn:aws:omics:us-west-2:555555555555:referenceStore/3242349265",
            "name": "MyReferenceStore",
           "creationTime": "2022-07-01T20:58:42.878Z"
        }
    ]
}
```

Dopo aver creato un archivio di riferimento, potete creare processi di importazione per caricare file di riferimento genomici al suo interno. A tale scopo, è necessario utilizzare o creare un ruolo IAM per accedere ai dati. Di seguito è riportata una policy di esempio.

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "s3:GetObject",
                "s3:GetBucketLocation"

            ],
            "Resource": [
                "arn:aws:s3:::amzn-s3-demo-bucket1",
                "arn:aws:s3:::amzn-s3-demo-bucket1/*"
            ]
        }
    ]
}
```

È inoltre necessario disporre di una politica di fiducia simile all'esempio seguente.

JSON

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

```
            "Effect": "Allow",
            "Principal": {
                "Service": [
                   "omics.amazonaws.com"
                ]
            },
            "Action": "sts:AssumeRole"
        }
    ]
}
```

Ora puoi importare un genoma di riferimento. Questo esempio utilizza Genome Reference Consortium Human Build 38 (hg38), che è ad accesso aperto e disponibile nel Registry of Open Data su. AWS Il bucket che ospita questi dati ha sede negli Stati Uniti orientali (Ohio). Per utilizzare i bucket in altre AWS regioni, puoi copiare i dati in un bucket Amazon S3 ospitato nella tua regione. Usa il seguente AWS CLI comando per copiare il genoma nel tuo bucket Amazon S3.

```
aws s3 cp s3://broad-references/hg38/v0/Homo_sapiens_assembly38.fasta s3://amzn-s3-demo-bucket
```

Puoi quindi iniziare il processo di importazione. Sostituisci *reference store ID* e *source file path* con il tuo input. *role ARN*

```
aws omics start-reference-import-job --reference-store-id reference store ID --role-arn role ARN --sources source file path
```

Dopo l'importazione dei dati, riceverai la seguente risposta in JSON.

```
{
        "id": "7252016478",
        "referenceStoreId": "3242349265",
        "roleArn": "arn:aws:iam::111122223333:role/OmicsReferenceImport",
        "status": "CREATED",
        "creationTime": "2022-07-01T21:15:13.727Z"
}
```

È possibile monitorare lo stato di un lavoro utilizzando il comando seguente. Nell'esempio seguente, sostituisci *reference store ID* e *job ID* con il tuo ID del negozio di riferimento e l'ID del lavoro su cui desideri saperne di più.

```
aws omics get-reference-import-job --reference-store-id reference store ID --id job ID
```

In risposta, riceverai una risposta con i dettagli relativi all'archivio di riferimento e al relativo stato.

```
{
    "id": "7252016478",
    "referenceStoreId": "3242349265",
    "roleArn": "arn:aws:iam::555555555555:role/OmicsReferenceImport",
    "status": "RUNNING",
    "creationTime": "2022-07-01T21:15:13.727Z",
    "sources": [
        {
            "sourceFile": "s3://amzn-s3-demo-bucket/Homo_sapiens_assembly38.fasta",
            "status": "IN_PROGRESS",
            "name": "MyReference"
        }
    ]
}
```

È inoltre possibile trovare il riferimento importato elencando i riferimenti e filtrandoli in base al nome del riferimento. Sostituiscilo *reference store ID* con il tuo ID del negozio di riferimento e aggiungi un filtro opzionale per restringere l'elenco.

```
aws omics list-references --reference-store-id reference store ID --filter
 name=MyReference
```

In risposta, riceverai le seguenti informazioni.

```
{
    "references": [
        {
            "id": "1234567890",
            "arn": "arn:aws:omics:us-west-2:555555555555:referenceStore/1234567890/
reference/1234567890",
            "referenceStoreId": "12345678",
            "md5": "7ff134953dcca8c8997453bbb80b6b5e",
            "status": "ACTIVE",
            "name": "MyReference",
            "creationTime": "2022-07-02T00:15:19.787Z",
            "updateTime": "2022-07-02T00:15:19.787Z"
```

```
        }
    ]
}
```

Per saperne di più sui metadati di riferimento, utilizza l'operazione get-reference-metadataAPI. Nell'esempio seguente, sostituiscilo *reference store ID* con il tuo ID del negozio di riferimento e *reference ID* con l'ID di riferimento su cui desideri saperne di più.

```
aws omics get-reference-metadata --reference-store-id reference store ID --id reference
 ID
```

Riceverai le seguenti informazioni in risposta.

```
{
    "id": "1234567890",
    "arn": "arn:aws:omics:us-west-2:555555555555:referenceStore/referencestoreID/
reference/referenceID",
    "referenceStoreId": "1234567890",
    "md5": "7ff134953dcca8c8997453bbb80b6b5e",
    "status": "ACTIVE",
    "name": "MyReference",
    "creationTime": "2022-07-02T00:15:19.787Z",
    "updateTime": "2022-07-02T00:15:19.787Z",
    "files": {
        "source": {
            "totalParts": 31,
            "partSize": 104857600,
            "contentLength": 3249912778
        },
        "index": {
            "totalParts": 1,
            "partSize": 104857600,
            "contentLength": 160928
        }
    }
}
```

È inoltre possibile scaricare parti del file di riferimento utilizzando get-reference. Nell'esempio seguente, sostituiscilo *reference store ID* con il tuo ID del negozio di riferimento e *reference ID* con l'ID di riferimento da cui desideri effettuare il download.

```
aws omics get-reference --reference-store-id reference store ID --id reference ID --
part-number 1 outfile.fa
```

Creazione di un archivio HealthOmics di sequenze

HealthOmics gli archivi di sequenze supportano l'archiviazione di file genomici nei formati non allineati di FASTQ (solo gzip) e. uBAM Supporta anche i formati allineati di e. BAM CRAM

I file importati vengono archiviati come set di lettura. Puoi aggiungere tag ai set di lettura e utilizzare le policy IAM per controllare l'accesso ai set di lettura. I set di lettura allineati richiedono un genoma di riferimento per allineare le sequenze genomiche, ma è facoltativo per i set di lettura non allineati.

Per memorizzare i set di lettura, devi prima creare un archivio di sequenze. Quando crei un archivio di sequenze, puoi specificare un bucket Amazon S3 opzionale come posizione di riserva e la posizione in cui vengono archiviati i log di accesso S3. La posizione di fallback viene utilizzata per archiviare tutti i file che non riescono a creare un set di lettura durante un caricamento diretto. Le posizioni di riserva sono disponibili per i sequence store creati dopo il 15 maggio 2023. La posizione di fallback viene specificata quando si crea l'archivio delle sequenze.

È possibile specificare fino a cinque chiavi di tag read set. Quando crei o aggiorni un set di lettura con una chiave di tag che corrisponde a una di queste chiavi, i tag del set di lettura vengono propagati all'oggetto Amazon S3 corrispondente. I tag di sistema creati da HealthOmics vengono propagati per impostazione predefinita.

Argomenti

Creazione di un archivio di sequenze utilizzando la console

Per creare un archivio di sequenze

1. Apri la HealthOmics console.

2. Se necessario, aprite il pannello di navigazione a sinistra (≡). Scegli i negozi Sequence.
3. Nella pagina Crea archivio sequenze, fornisci le seguenti informazioni

   - Sequence Store name: un nome univoco per questo negozio.
   - Descrizione (opzionale): una descrizione di questo archivio di sequenze.
4. Per una posizione di fallback in S3, specifica una posizione Amazon S3. HealthOmics utilizza la posizione di fallback per archiviare tutti i file che non riescono a creare un set di lettura durante un caricamento diretto. È necessario concedere al HealthOmics servizio l'accesso in scrittura alla posizione di fallback di Amazon S3. Per un esempio di policy, consulta Configura una posizione di fallback.

   Le posizioni di riserva non sono disponibili per i sequence store creati prima del 16 maggio 2023.
5. (Facoltativo) Per le chiavi dei tag Read set per la propagazione S3, puoi inserire fino a cinque chiavi di lettura per propagarle da un set di lettura agli oggetti S3 sottostanti. Propagando i tag da un set di lettura all'oggetto S3, puoi concedere autorizzazioni di accesso a S3 basate sui tag agli utenti and/or finali per visualizzare i tag propagati tramite l'operazione dell'API Amazon S3. getObjectTagging

   a. Inserisci un valore chiave nella casella di testo. La console crea una nuova casella di testo per aggiungere la chiave successiva.
   b. (Facoltativo) Scegliete Rimuovi per rimuovere tutte le chiavi.
6. In Crittografia dei dati, seleziona se desideri che la crittografia dei dati sia di proprietà e gestita da AWS o utilizzi una CMK gestita dal cliente.
7. (Facoltativo) In S3 Data access, seleziona se creare un nuovo ruolo e una nuova policy per accedere al Sequence Store tramite Amazon S3.
8. (Facoltativo) Per la registrazione degli accessi S3, seleziona `Enabled` se desideri che Amazon S3 raccolga i record dei log di accesso.

   Per la posizione di registrazione di Access in S3, specifica una posizione Amazon S3 in cui archiviare i log. Questo campo è visibile solo se hai abilitato la registrazione degli accessi S3.
9. Tag (opzionale): fornisci fino a 50 tag per questo archivio di sequenze. Questi tag sono separati dai tag del set di lettura che vengono impostati durante l' import/tag aggiornamento del set di lettura

Dopo aver creato il negozio, è pronto perImportazione di file genomici.

Creazione di un archivio di sequenze utilizzando la CLI

Nell'esempio seguente, sostituiscilo *sequence store name* con il nome che hai scelto per il tuo archivio di sequenze.

```
aws omics create-sequence-store --name sequence store name --fallback-location "s3://
amzn-s3-demo-bucket"
```

Riceverai la seguente risposta in JSON, che include il numero ID del tuo sequence store appena creato.

```
{
    "id": "3936421177",
    "arn": "arn:aws:omics:us-west-2:111122223333:sequenceStore/3936421177",
    "name": "sequence_store_example_name",
    "creationTime": "2022-07-13T20:09:26.038Z"
    "fallbackLocation" : "s3://amzn-s3-demo-bucket"
}
```

È inoltre possibile visualizzare tutti gli archivi di sequenze associati al proprio account utilizzando il list-sequence-storescomando, come illustrato di seguito.

```
aws omics list-sequence-stores
```

Riceverai la seguente risposta.

```
{
    "sequenceStores": [
        {
            "arn": "arn:aws:omics:us-west-2:111122223333:sequenceStore/3936421177",
            "id": "3936421177",
            "name": "MySequenceStore",
            "creationTime": "2022-07-13T20:09:26.038Z",
            "updatedTime": "2024-09-13T04:11:31.242Z",
            "fallbackLocation" : "s3://amzn-s3-demo-bucket",
            "status": "Active"
        }
    ]
}
```

È possibile utilizzarla get-sequence-storeper saperne di più su un archivio di sequenze utilizzando il relativo ID, come illustrato nell'esempio seguente:

```
aws omics get-sequence-store --id sequence store ID
```

Riceverai la seguente risposta:

```
{
  "arn": "arn:aws:omics:us-west-2:123456789012:sequenceStore/sequencestoreID",
  "creationTime": "2024-01-12T04:45:29.857Z",
  "updatedTime": "2024-09-13T04:11:31.242Z",
  "description": null,
  "fallbackLocation": null,
  "id": "2015356892",
  "name": "MySequenceStore",
  "s3Access": {
      "s3AccessPointArn": "arn:aws:s3:us-
west-2:123456789012:accesspoint/592761533288-2015356892",
      "s3Uri": "s3://592761533288-2015356892-ajdpi90jdas90a79fh9a8ja98jdfa9jf98-
s3alias/592761533288/sequenceStore/2015356892/",
      "accessLogLocation": "s3://IAD-seq-store-log/2015356892/"
  },
  "sseConfig": {
      "keyArn": "arn:aws:kms:us-west-2:123456789012:key/eb2b30f5-635d-4b6d-b0f9-
d3889fe0e648",
      "type": "KMS"
  },
  "status": "Active",
  "statusMessage": null,
  "setTagsToSync": ["withdrawn","protocol"],
}
```

Dopo la creazione, è possibile aggiornare anche diversi parametri del negozio. Questa operazione può essere eseguita tramite la console o l'`updateSequenceStore`operazione API.

Aggiornamento di un archivio di sequenze

Per aggiornare un archivio di sequenze, effettuate le seguenti operazioni:

1. Apri la HealthOmics console.
2. Se necessario, aprite il riquadro di navigazione a sinistra (≡). Scegli i negozi Sequence.

3. Scegli l'archivio delle sequenze da aggiornare.
4. Nel pannello Dettagli, scegliete Modifica.
5. Nella pagina Modifica dettagli, puoi aggiornare i seguenti campi:

   - Nome del negozio in sequenza: un nome univoco per questo negozio.
   - Descrizione: una descrizione di questo archivio di sequenze.
   - Posizione di fallback in S3, specifica una posizione Amazon S3. HealthOmics utilizza la posizione di fallback per archiviare tutti i file che non riescono a creare un set di lettura durante un caricamento diretto.
   - Leggi le chiavi dei tag del set di lettura per la propagazione S3, puoi inserire fino a cinque chiavi del set di lettura da propagare su Amazon S3.
   - (Facoltativo) Per la registrazione degli accessi S3, seleziona `Enabled` se desideri che Amazon S3 raccolga i record dei log di accesso.

     Per la posizione di registrazione di Access in S3, specifica una posizione Amazon S3 in cui archiviare i log. Questo campo è visibile solo se hai abilitato la registrazione degli accessi S3.
   - Tag (opzionale): fornisci fino a 50 tag per questo archivio di sequenze.

Aggiornamento dei tag di lettura per un archivio di sequenze

Per aggiornare i tag del set di lettura o altri campi per un archivio di sequenze, procedi nel seguente modo:

1. Apri la HealthOmics console.
2. Se necessario, apri il pannello di navigazione a sinistra (≡). Scegli i negozi Sequence.
3. Scegli l'archivio di sequenze che desideri aggiornare.
4. Seleziona la scheda Details (Dettagli).
5. Scegli Modifica.
6. Aggiungi nuovi tag di lettura o elimina i tag esistenti, se necessario.
7. Aggiorna il nome, la descrizione, la posizione di riserva o l'accesso ai dati S3, come richiesto.
8. Scegli Save changes (Salva modifiche).

Importazione di file genomici

Per importare file genomici in un archivio di sequenze, procedi nel seguente modo:

Per importare un file genomico

1. Apri la HealthOmics console.
2. Se necessario, apri il pannello di navigazione a sinistra (≡). Scegli Sequence stores.
3. Nella pagina Sequence stores, scegli l'archivio di sequenze in cui vuoi importare i tuoi file.
4. Nella pagina dell'archivio delle singole sequenze, scegli Importa file genomici.
5. Nella pagina Specificare i dettagli di importazione, fornite le seguenti informazioni
   - Ruolo IAM: il ruolo IAM che può accedere ai file genomici su Amazon S3.
   - Genoma di riferimento: il genoma di riferimento per questi dati genomici.
6. Nella pagina Specificare il manifesto di importazione, specificare il seguente file manifesto di informazioni. Il file manifest è un file JSON o YAML che descrive le informazioni essenziali dei dati genomici. Per informazioni sul file manifest, vedere. Importazione di set di lettura in un archivio di HealthOmics sequenze
7. Fate clic su Crea processo di importazione.

Eliminazione di archivi HealthOmics di riferimenti e sequenze

È possibile eliminare sia gli archivi di riferimento che quelli di sequenza. Gli archivi di sequenze possono essere eliminati solo se non contengono set di lettura e gli archivi di riferimento possono essere eliminati solo se non contengono riferimenti. L'eliminazione di una sequenza o di un archivio di riferimenti elimina anche tutti i tag associati a tale archivio.

L'esempio seguente mostra come eliminare un archivio di riferimenti utilizzando. AWS CLI Se l'azione ha esito positivo, non riceverai alcuna risposta. Nell'esempio seguente, sostituiscilo *reference store ID* con il tuo ID del negozio di riferimento.

```
aws omics delete-reference-store --id reference store ID
```

L'esempio seguente mostra come eliminare un archivio di sequenze. Non si riceve una risposta se l'azione ha esito positivo. Nell'esempio seguente, sostituiscilo *sequence store ID* con il tuo Sequence Store ID.

```
aws omics delete-sequence-store --id sequence store ID
```

È inoltre possibile eliminare un riferimento in un archivio di riferimenti, come illustrato nell'esempio seguente. I riferimenti possono essere eliminati solo se non vengono utilizzati in un set di lettura, in un archivio di varianti o in un archivio di annotazioni. Nell'esempio seguente, sostituiscilo *reference store ID* con l'ID del tuo negozio di riferimento e sostituiscilo *reference ID* con l'ID del riferimento che desideri eliminare.

```
aws omics delete-reference  --id reference ID --reference-store-id reference store ID
```

Importazione di set di lettura in un archivio di HealthOmics sequenze

Dopo aver creato l'archivio delle sequenze, create processi di importazione per caricare i set di lettura nell'archivio dati. Puoi caricare i tuoi file da un bucket Amazon S3 oppure caricarli direttamente utilizzando le operazioni API sincrone. Il tuo bucket Amazon S3 deve trovarsi nella stessa regione del tuo Sequence Store.

Puoi caricare qualsiasi combinazione di set di lettura allineati e non allineati nel tuo archivio di sequenze, tuttavia, se uno dei set di lettura nell'importazione è allineato, devi includere un genoma di riferimento.

Puoi riutilizzare la policy di accesso IAM che hai usato per creare l'archivio di riferimento.

I seguenti argomenti descrivono i passaggi principali da seguire per importare un set di lettura nel proprio Sequence Store e quindi ottenere informazioni sui dati importati.

Argomenti

- Caricare file su Amazon S3
- Creazione di un file manifesto
- Avvio del processo di importazione
- Monitora il processo di importazione
- Trovate i file di sequenza importati
- Ottieni dettagli su un set di lettura
- Scarica i file di dati del set di lettura

Caricare file su Amazon S3

L'esempio seguente mostra come spostare i file nel bucket Amazon S3.

```
aws s3 cp s3://1000genomes/phase1/data/HG00100/alignment/
HG00100.chrom20.ILLUMINA.bwa.GBR.low_coverage.20101123.bam s3://your-bucket
aws s3 cp s3://1000genomes/phase3/data/HG00146/sequence_read/SRR233106_1.filt.fastq.gz
 s3://your-bucket
aws s3 cp s3://1000genomes/phase3/data/HG00146/sequence_read/SRR233106_2.filt.fastq.gz
 s3://your-bucket
aws s3 cp s3://1000genomes/data/HG00096/alignment/
HG00096.alt_bwamem_GRCh38DH.20150718.GBR.low_coverage.cram s3://your-bucket
aws s3 cp s3://gatk-test-data/wgs_ubam/NA12878_20k/NA12878_A.bam s3://your-bucket
```

L'esempio BAM e quello CRAM utilizzato in questo esempio richiedono riferimenti genomici diversi, e. Hg19 Hg38 Per saperne di più o per accedere a questi riferimenti, vedere The Broad Genome References nel Registry of Open Data su. AWS

Creazione di un file manifesto

È inoltre necessario creare un file manifest in JSON in cui modellare il processo di importazione `import.json` (vedere l'esempio seguente). Se create un archivio di sequenze nella console, non è necessario specificare `sequenceStoreId` o`roleARN`, quindi il file manifest inizia con l'`sources`input.

API manifest

L'esempio seguente importa tre set di lettura utilizzando l'API: uno FASTQBAM, uno e unoCRAM.

```
{
  "sequenceStoreId": "3936421177",
  "roleArn": "arn:aws:iam::555555555555:role/OmicsImport",
  "sources":
  [
      {
          "sourceFiles":
          {
              "source1": "s3://amzn-s3-demo-bucket/
HG00100.chrom20.ILLUMINA.bwa.GBR.low_coverage.20101123.bam"
          },
          "sourceFileType": "BAM",
```

```
          "subjectId": "mySubject",
          "sampleId": "mySample",
          "referenceArn": "arn:aws:omics:us-
west-2:555555555555:referenceStore/0123456789/reference/0000000001",
          "name": "HG00100",
          "description": "BAM for HG00100",
          "generatedFrom": "1000 Genomes"
      },
      {
          "sourceFiles":
          {
              "source1": "s3://amzn-s3-demo-bucket/SRR233106_1.filt.fastq.gz",
              "source2": "s3://amzn-s3-demo-bucket/SRR233106_2.filt.fastq.gz"
          },
          "sourceFileType": "FASTQ",
          "subjectId": "mySubject",
          "sampleId": "mySample",
          // NOTE: there is no reference arn required here
          "name": "HG00146",
          "description": "FASTQ for HG00146",
          "generatedFrom": "1000 Genomes"
      },
      {
          "sourceFiles":
          {
              "source1": "s3://amzn-s3-demo-bucket/
HG00096.alt_bwamem_GRCh38DH.20150718.GBR.low_coverage.cram"
          },
          "sourceFileType": "CRAM",
          "subjectId": "mySubject",
          "sampleId": "mySample",
          "referenceArn": "arn:aws:omics:us-
west-2:555555555555:referenceStore/0123456789/reference/0000000001",
          "name": "HG00096",
          "description": "CRAM for HG00096",
          "generatedFrom": "1000 Genomes"
      },
      {
          "sourceFiles":
          {
              "source1": "s3://amzn-s3-demo-bucket/NA12878_A.bam"
          },
          "sourceFileType": "UBAM",
          "subjectId": "mySubject",
```

```
          "sampleId": "mySample",
          // NOTE: there is no reference arn required here
          "name": "NA12878_A",
          "description": "uBAM for NA12878",
          "generatedFrom": "GATK Test Data"
      }
  ]
}
```

Console manifest

Questo codice di esempio viene utilizzato per importare un singolo set di lettura utilizzando la console.

```
[
  {
      "sourceFiles":
      {
          "source1": "s3://amzn-s3-demo-bucket/
HG00100.chrom20.ILLUMINA.bwa.GBR.low_coverage.20101123.bam"
      },
      "sourceFileType": "BAM",
      "subjectId": "mySubject",
      "sampleId": "mySample",
      "name": "HG00100",
      "description": "BAM for HG00100",
      "generatedFrom": "1000 Genomes"
  },
  {
      "sourceFiles":
      {
          "source1": "s3://amzn-s3-demo-bucket/SRR233106_1.filt.fastq.gz",
          "source2": "s3://amzn-s3-demo-bucket/SRR233106_2.filt.fastq.gz"
      },
      "sourceFileType": "FASTQ",
      "subjectId": "mySubject",
      "sampleId": "mySample",
      "name": "HG00146",
      "description": "FASTQ for HG00146",
      "generatedFrom": "1000 Genomes"
  },
  {
      "sourceFiles":
```

```
      {
          "source1": "s3://your-bucket/
HG00096.alt_bwamem_GRCh38DH.20150718.GBR.low_coverage.cram"
      },
      "sourceFileType": "CRAM",
      "subjectId": "mySubject",
      "sampleId": "mySample",
      "name": "HG00096",
      "description": "CRAM for HG00096",
      "generatedFrom": "1000 Genomes"
  },
  {
      "sourceFiles":
      {
          "source1": "s3://amzn-s3-demo-bucket/NA12878_A.bam"
      },
      "sourceFileType": "UBAM",
      "subjectId": "mySubject",
      "sampleId": "mySample",
      "name": "NA12878_A",
      "description": "uBAM for NA12878",
      "generatedFrom": "GATK Test Data"
  }
]
```

In alternativa, puoi caricare il file manifest in formato YAML.

Avvio del processo di importazione

Per avviare il processo di importazione, utilizzare il AWS CLI comando seguente.

```
aws omics start-read-set-import-job --cli-input-json file://import.json
```

Riceverai la seguente risposta, che indica che la creazione di posti di lavoro è riuscita.

```
{
  "id": "3660451514",
  "sequenceStoreId": "3936421177",
  "roleArn": "arn:aws:iam::111122223333:role/OmicsImport",
  "status": "CREATED",
  "creationTime": "2022-07-13T22:14:59.309Z"
```

```
}
```

Monitora il processo di importazione

Dopo l'avvio del processo di importazione, è possibile monitorarne l'avanzamento con il seguente comando. Nell'esempio seguente, sostituitelo *sequence store id* con il vostro Sequence Store ID e sostituitelo *job import ID* con l'ID di importazione.

```
aws omics get-read-set-import-job --sequence-store-id sequence store id --id job import
 ID
```

Di seguito vengono illustrati gli stati di tutti i lavori di importazione associati all'ID dell'archivio delle sequenze specificato.

```
{
  "id": "1234567890",
  "sequenceStoreId": "1234567890",
  "roleArn": "arn:aws:iam::111122223333:role/OmicsImport",
  "status": "RUNNING",
  "statusMessage": "The job is currently in progress.",
  "creationTime": "2022-07-13T22:14:59.309Z",
  "sources": [
      {
          "sourceFiles":
          {
              "source1": "s3://amzn-s3-demo-bucket/
HG00100.chrom20.ILLUMINA.bwa.GBR.low_coverage.20101123.bam"
          },
          "sourceFileType": "BAM",
          "status": "IN_PROGRESS",
          "statusMessage": "The job is currently in progress."
          "subjectId": "mySubject",
          "sampleId": "mySample",
          "referenceArn": "arn:aws:omics:us-
west-2:111122223333:referenceStore/3242349265/reference/8625408453",
          "name": "HG00100",
          "description": "BAM for HG00100",
          "generatedFrom": "1000 Genomes",
          "readSetID": "1234567890"
      },
      {
          "sourceFiles":
```

```
          {
              "source1": "s3://amzn-s3-demo-bucket/SRR233106_1.filt.fastq.gz",
              "source2": "s3://amzn-s3-demo-bucket/SRR233106_2.filt.fastq.gz"
          },
          "sourceFileType": "FASTQ",
          "status": "IN_PROGRESS",
          "statusMessage": "The job is currently in progress."
          "subjectId": "mySubject",
          "sampleId": "mySample",
          "name": "HG00146",
          "description": "FASTQ for HG00146",
          "generatedFrom": "1000 Genomes",
          "readSetID": "1234567890"
      },
      {
          "sourceFiles":
          {
              "source1": "s3://amzn-s3-demo-bucket/
HG00096.alt_bwamem_GRCh38DH.20150718.GBR.low_coverage.cram"
          },
          "sourceFileType": "CRAM",
          "status": "IN_PROGRESS",
          "statusMessage": "The job is currently in progress."
          "subjectId": "mySubject",
          "sampleId": "mySample",
          "referenceArn": "arn:aws:omics:us-
west-2:111122223333:referenceStore/3242349265/reference/1234568870",
          "name": "HG00096",
          "description": "CRAM for HG00096",
          "generatedFrom": "1000 Genomes",
          "readSetID": "1234567890"
      },
      {
          "sourceFiles":
          {
              "source1": "s3://amzn-s3-demo-bucket/NA12878_A.bam"
          },
          "sourceFileType": "UBAM",
          "status": "IN_PROGRESS",
          "statusMessage": "The job is currently in progress."
          "subjectId": "mySubject",
          "sampleId": "mySample",
          "name": "NA12878_A",
          "description": "uBAM for NA12878",
```

```
          "generatedFrom": "GATK Test Data",
          "readSetID": "1234567890"
      }
  ]
}
```

Trovate i file di sequenza importati

Una volta completato il lavoro, potete utilizzare l'operazione list-read-setsAPI per trovare i file di sequenza importati. Nell'esempio seguente, sostituiscilo *sequence store id* con il tuo Sequence Store ID.

```
aws omics list-read-sets --sequence-store-id sequence store id
```

Riceverai la seguente risposta.

```
{
  "readSets": [
      {
          "id": "0000000001",
          "arn": "arn:aws:omics:us-west-2:111122223333:sequenceStore/01234567890/
readSet/0000000001",
          "sequenceStoreId": "1234567890",
          "subjectId": "mySubject",
          "sampleId": "mySample",
          "status": "ACTIVE",
          "name": "HG00100",
          "description": "BAM for HG00100",
          "referenceArn": "arn:aws:omics:us-
west-2:111122223333:referenceStore/01234567890/reference/0000000001",
          "fileType": "BAM",
          "sequenceInformation": {
              "totalReadCount": 9194,
              "totalBaseCount": 928594,
              "generatedFrom": "1000 Genomes",
              "alignment": "ALIGNED"
          },
          "creationTime": "2022-07-13T23:25:20Z"
          "creationType": "IMPORT",
          "etag": {
              "algorithm": "BAM_MD5up",
              "source1": "d1d65429212d61d115bb19f510d4bd02"
```

```
            }
        },
        {
            "id": "0000000002",
            "arn": "arn:aws:omics:us-west-2:111122223333:sequenceStore/0123456789/
readSet/0000000002",
            "sequenceStoreId": "0123456789",
            "subjectId": "mySubject",
            "sampleId": "mySample",
            "status": "ACTIVE",
            "name": "HG00146",
            "description": "FASTQ for HG00146",
            "fileType": "FASTQ",
            "sequenceInformation": {
                "totalReadCount": 8000000,
                "totalBaseCount": 1184000000,
                "generatedFrom": "1000 Genomes",
                "alignment": "UNALIGNED"
            },
            "creationTime": "2022-07-13T23:26:43Z"
            "creationType": "IMPORT",
            "etag": {
                "algorithm": "FASTQ_MD5up",
                "source1": "ca78f685c26e7cc2bf3e28e3ec4d49cd"
            }
        },
        {
            "id": "0000000003",
            "arn": "arn:aws:omics:us-west-2:111122223333:sequenceStore/0123456789/
readSet/0000000003",
            "sequenceStoreId": "0123456789",
            "subjectId": "mySubject",
            "sampleId": "mySample",
            "status": "ACTIVE",
            "name": "HG00096",
            "description": "CRAM for HG00096",
            "referenceArn": "arn:aws:omics:us-
west-2:111122223333:referenceStore/0123456789/reference/0000000001",
            "fileType": "CRAM",
            "sequenceInformation": {
                "totalReadCount": 85466534,
                "totalBaseCount": 24000004881,
                "generatedFrom": "1000 Genomes",
                "alignment": "ALIGNED"
```

```
            },
            "creationTime": "2022-07-13T23:30:41Z"
            "creationType": "IMPORT",
            "etag": {
                "algorithm": "CRAM_MD5up",
                "source1": "66817940f3025a760e6da4652f3e927e"
            }
        },
        {
            "id": "0000000004",
            "arn": "arn:aws:omics:us-west-2:111122223333:sequenceStore/0123456789/
readSet/0000000004",
            "sequenceStoreId": "0123456789",
            "subjectId": "mySubject",
            "sampleId": "mySample",
            "status": "ACTIVE",
            "name": "NA12878_A",
            "description": "uBAM for NA12878",
            "fileType": "UBAM",
            "sequenceInformation": {
                "totalReadCount": 20000,
                "totalBaseCount": 5000000,
                "generatedFrom": "GATK Test Data",
                "alignment": "ALIGNED"
            },
            "creationTime": "2022-07-13T23:30:41Z"
            "creationType": "IMPORT",
            "etag": {
                "algorithm": "BAM_MD5up",
                "source1": "640eb686263e9f63bcda12c35b84f5c7"
            }
        }
    ]
}
```


Ottieni dettagli su un set di lettura

Per visualizzare maggiori dettagli su un set di lettura, utilizza l'operazione GetReadSetMetadataAPI. Nell'esempio seguente, sostituiscilo *`sequence store id`* con il tuo Sequence Store ID e sostituiscilo *`read set id`* con il tuo ID del set di lettura.

```
aws omics get-read-set-metadata --sequence-store-id sequence store id --id read set id
```

Riceverai la seguente risposta.


```
{
"arn": "arn:aws:omics:us-west-2:123456789012:sequenceStore/2015356892/
readSet/9515444019",
"creationTime": "2024-01-12T04:50:33.548Z",
"creationType": "IMPORT",
"creationJobId": "33222111",
"description": null,
"etag": {
  "algorithm": "FASTQ_MD5up",
  "source1": "00d0885ba3eeb211c8c84520d3fa26ec",
  "source2": "00d0885ba3eeb211c8c84520d3fa26ec"
},
"fileType": "FASTQ",
"files": {
  "index": null,
  "source1": {
    "contentLength": 10818,
    "partSize": 104857600,
    "s3Access": {
      "s3Uri": "s3://accountID-sequence store ID-ajdpi90jdas90a79fh9a8ja98jdfa9jf98-
s3alias/592761533288/sequenceStore/2015356892/readSet/9515444019/
import_source1.fastq.gz"
},
    "totalParts": 1
  },
  "source2": {
    "contentLength": 10818,
    "partSize": 104857600,
    "s3Access": {
      "s3Uri": "s3://accountID-sequence store ID-ajdpi90jdas90a79fh9a8ja98jdfa9jf98-
s3alias/592761533288/sequenceStore/2015356892/readSet/9515444019/
import_source1.fastq.gz"
    },
    "totalParts": 1
  }
},
"id": "9515444019",
"name": "paired-fastq-import",
```

```
"sampleId": "sampleId-paired-fastq-import",
"sequenceInformation": {
  "alignment": "UNALIGNED",
  "generatedFrom": null,
  "totalBaseCount": 30000,
  "totalReadCount": 200
},
"sequenceStoreId": "2015356892",
"status": "ACTIVE",
"statusMessage": null,
"subjectId": "subjectId-paired-fastq-import"
}
```

Scarica i file di dati del set di lettura

Puoi accedere agli oggetti per un set di lettura attivo utilizzando l'operazione dell'GetObjectAPI Amazon S3. L'URI dell'oggetto viene restituito nella risposta dell'GetReadSetMetadataAPI. Per ulteriori informazioni, consulta Accesso ai set di HealthOmics lettura con Amazon S3 URIs.

In alternativa, utilizzate l'operazione HealthOmics GetReadSet API. È possibile GetReadSet utilizzare il download in parallelo scaricando singole parti. Queste parti sono simili alle parti di Amazon S3. Di seguito è riportato un esempio di come scaricare la parte 1 da un set di lettura. Nell'esempio seguente, sostituiscilo *`sequence store id`* con il tuo Sequence Store ID e sostituiscilo *`read set id`* con il tuo ID del set di lettura.

```
aws omics get-read-set --sequence-store-id sequence store id --id read set id  --part-
number 1 outfile.bam
```

È inoltre possibile utilizzare HealthOmics Transfer Manager per scaricare file da utilizzare come HealthOmics riferimento o come set di lettura. Puoi scaricare il HealthOmics Transfer Manager qui. Per ulteriori informazioni sull'uso e la configurazione di Transfer Manager, consulta questo GitHubRepository.

Caricamento diretto su un archivio HealthOmics di sequenze

Ti consigliamo di utilizzare HealthOmics Transfer Manager per aggiungere file al tuo archivio di sequenze. Per ulteriori informazioni sull'utilizzo di Transfer Manager, consultate questo GitHubRepository. Puoi anche caricare i tuoi set di lettura direttamente in un archivio di sequenze tramite le operazioni dell'API di caricamento diretto.

I set di lettura per il caricamento diretto esistono per primi nello `PROCESSING_UPLOAD` stato. Ciò significa che le parti del file sono attualmente in fase di caricamento ed è possibile accedere ai metadati del set di lettura. Dopo il caricamento delle parti e la convalida dei checksum, il set di lettura diventa `ACTIVE` e si comporta come un set di lettura importato.

Se il caricamento diretto fallisce, lo stato del set di lettura viene visualizzato come. `UPLOAD_FAILED` Puoi configurare un bucket Amazon S3 come posizione di riserva per i file che non vengono caricati. Le posizioni di riserva sono disponibili per i sequence store creati dopo il 15 maggio 2023.

Argomenti

Caricamento diretto in un archivio di sequenze utilizzando AWS CLI

Per iniziare, avvia un caricamento in più parti. È possibile eseguire questa operazione utilizzando AWS CLI, come illustrato nell'esempio seguente.

Per il caricamento diretto utilizzando AWS CLI i comandi

1. Create le parti separando i dati, come illustrato nell'esempio seguente.

   ```
   split -b 100MiB SRR233106_1.filt.fastq.gz source1_part_
   ```

2. Dopo che i file sorgente sono stati suddivisi in parti, create un caricamento del set di lettura in più parti, come illustrato nell'esempio seguente. Sostituite *`sequence store ID`* e gli altri parametri con l'ID del vostro archivio di sequenza e altri valori.

   ```
   aws omics create-multipart-read-set-upload \
   --sequence-store-id sequence store ID \
   --name upload name \
   --source-file-type FASTQ \
   --subject-id subject ID \
   --sample-id sample ID \
   --description "FASTQ for HG00146" "description of upload" \
   --generated-from "1000 Genomes""source of imported files"
   ```

   Ottieni i metadati `uploadID` e gli altri metadati nella risposta. Utilizza il `uploadID` per la fase successiva del processo di caricamento.

```
{
"sequenceStoreId": "1504776472",
"uploadId": "7640892890",
"sourceFileType": "FASTQ",
"subjectId": "mySubject",
"sampleId": "mySample",
"generatedFrom": "1000 Genomes",
"name": "HG00146",
"description": "FASTQ for HG00146",
"creationTime": "2023-11-20T23:40:47.437522+00:00"
}
```

3. Aggiungi i tuoi set di lettura al caricamento. Se il file è sufficientemente piccolo, è sufficiente eseguire questo passaggio una sola volta. Per file di grandi dimensioni, esegui questo passaggio per ogni parte del file. Se si carica una nuova parte utilizzando un numero di parte utilizzato in precedenza, la parte caricata in precedenza viene sovrascritta.

   Nell'esempio seguente *sequence store IDupload ID*, sostituite e gli altri parametri con i vostri valori.

   ```
   aws omics upload-read-set-part \
   --sequence-store-id sequence store ID \
   --upload-id upload ID \
   --part-source SOURCE1 \
   --part-number part number \
   --payload  source1/source1_part_aa.fastq.gz
   ```

   La risposta è un ID che potete utilizzare per verificare che il file caricato corrisponda al file desiderato.

   ```
   {
   "checksum": "984979b9928ae8d8622286c4a9cd8e99d964a22d59ed0f5722e1733eb280e635"
   }
   ```

4. Continua a caricare le parti del file, se necessario. Per verificare che i set di lettura siano stati caricati, utilizzate l'operazione API list-read-set-upload-parts, come illustrato di seguito. Nell'esempio seguente *sequence store ID upload ID*, sostituisci «e» *part source* con il tuo input.

   ```
   aws omics list-read-set-upload-parts \
   ```

```
 --sequence-store-id sequence store ID \
 --upload-id upload ID \
 --part-source SOURCE1
```

La risposta restituisce il numero di set di lettura, la dimensione e il timestamp dell'ultimo aggiornamento.


```
{
"parts": [
    {
        "partNumber": 1,
        "partSize": 104857600,
        "partSource": "SOURCE1",
        "checksum": "MVMQk+vB9C3Ge8ADHkbKq752n3BCUzyl41qEkqlOD5M=",
        "creationTime": "2023-11-20T23:58:03.500823+00:00",
        "lastUpdatedTime": "2023-11-20T23:58:03.500831+00:00"
    },
    {
        "partNumber": 2,
        "partSize": 104857600,
        "partSource": "SOURCE1",
        "checksum": "keZzVzJNChAqgOdZMvOmjBwrOPM0enPj1UAfs0nvRto=",
        "creationTime": "2023-11-21T00:02:03.813013+00:00",
        "lastUpdatedTime": "2023-11-21T00:02:03.813025+00:00"
    },
    {
        "partNumber": 3,
        "partSize": 100339539,
        "partSource": "SOURCE1",
        "checksum": "TBkNfMsaeDpXzEf3ldlbi0ipFDPaohKHyZ+LF1J4CHk=",
        "creationTime": "2023-11-21T00:09:11.705198+00:00",
        "lastUpdatedTime": "2023-11-21T00:09:11.705208+00:00"
    }
]
}
```


5. Per visualizzare tutti i caricamenti attivi di set di lettura multiparte, utilizzate list-multipart-read-set-uploads, come illustrato di seguito. *sequence store ID*Sostituitelo con l'ID del vostro archivio di sequenze.

```
aws omics list-multipart-read-set-uploads --sequence-store-id
            sequence store ID
```

Questa API restituisce solo caricamenti di set di lettura multiparte in corso. Dopo che i set di lettura sono stati inseriti**`ACTIVE`**, o se il caricamento non è riuscito, il caricamento non verrà restituito nella risposta all'API -uploads. list-multipart-read-set Per visualizzare i set di lettura attivi, utilizza l'API. list-read-sets Di seguito è riportato un esempio di risposta per list-multipart-read-set-uploads.


```
{
"uploads": [
    {
        "sequenceStoreId": "1234567890",
        "uploadId": "8749584421",
        "sourceFileType": "FASTQ",
        "subjectId": "mySubject",
        "sampleId": "mySample",
        "generatedFrom": "1000 Genomes",
        "name": "HG00146",
        "description": "FASTQ for HG00146",
        "creationTime": "2023-11-29T19:22:51.349298+00:00"
    },
    {
        "sequenceStoreId": "1234567890",
        "uploadId": "5290538638",
        "sourceFileType": "BAM",
        "subjectId": "mySubject",
        "sampleId": "mySample",
        "generatedFrom": "1000 Genomes",
        "referenceArn": "arn:aws:omics:us-
west-2:123456789012:referenceStore/8168613728/reference/2190697383",
        "name": "HG00146",
        "description": "BAM for HG00146",
        "creationTime": "2023-11-29T19:23:33.116516+00:00"
    },
    {
        "sequenceStoreId": "1234567890",
        "uploadId": "4174220862",
        "sourceFileType": "BAM",
        "subjectId": "mySubject",
        "sampleId": "mySample",
        "generatedFrom": "1000 Genomes",
```

```
        "referenceArn": "arn:aws:omics:us-
west-2:123456789012:referenceStore/8168613728/reference/2190697383",
        "name": "HG00147",
        "description": "BAM for HG00147",
        "creationTime": "2023-11-29T19:23:47.007866+00:00"
    }
]
}
```

6. Dopo aver caricato tutte le parti del file, usa complete-multipart-read-set-upload per concludere il processo di caricamento, come mostrato nell'esempio seguente. Sostituisci *sequence store ID* e il parametro per le parti con i tuoi valori. *upload ID*

```
aws omics complete-multipart-read-set-upload \
--sequence-store-id sequence store ID \
--upload-id upload ID \
--parts '[{"checksum":"gaCBQMe+rpCFZxLpoP6gydBoXaKKDA/
Vobh5zBDb4W4=","partNumber":1,"partSource":"SOURCE1"}]'
```

La risposta per complete-multipart-read-set-upload è il set di lettura IDs per i set di lettura importati.

```
{
"readSetId": "0000000001"
}
```

7. Per interrompere il caricamento, usa abort-multipart-read-set-upload con l'ID di caricamento per terminare il processo di caricamento. Sostituisci *sequence store ID* e *upload ID* con i tuoi valori di parametro.

```
aws omics abort-multipart-read-set-upload \
--sequence-store-id sequence store ID \
--upload-id upload ID
```

8. Una volta completato il caricamento, recuperate i dati dal set di lettura utilizzando get-read-set, come illustrato di seguito. Se il caricamento è ancora in fase di elaborazione, get-read-setrestituisce metadati limitati e i file indice generati non sono disponibili. Sostituisci *sequence store ID* e gli altri parametri con il tuo input.

```
aws omics get-read-set
 --sequence-store-id sequence store ID \
```

```
 --id read set ID \
 --file SOURCE1 \
 --part-number 1 myfile.fastq.gz
```

9. Per controllare i metadati, incluso lo stato del caricamento, utilizza l'operazione get-read-set-metadataAPI.

```
aws omics get-read-set-metadata --sequence-store-id sequence store ID --id read set
 ID
```

La risposta include dettagli sui metadati come il tipo di file, l'ARN di riferimento, il numero di file e la lunghezza delle sequenze. Include anche lo stato. Gli stati possibili sono PROCESSING_UPLOADACTIVE, eUPLOAD_FAILED.

```
{
"id": "0000000001",
"arn": "arn:aws:omics:us-west-2:555555555555:sequenceStore/0123456789/
readSet/0000000001",
"sequenceStoreId": "0123456789",
"subjectId": "mySubject",
"sampleId": "mySample",
"status": "PROCESSING_UPLOAD",
"name": "HG00146",
"description": "FASTQ for HG00146",
"fileType": "FASTQ",
"creationTime": "2022-07-13T23:25:20Z",
"files": {
    "source1": {
        "totalParts": 5,
        "partSize": 123456789012,
        "contentLength": 6836725,

    },
    "source2": {
        "totalParts": 5,
        "partSize": 123456789056,
        "contentLength": 6836726
    }
},
'creationType": "UPLOAD"
}
```

Configura una posizione di fallback

Quando crei o aggiorni un archivio di sequenze, puoi configurare un bucket Amazon S3 come posizione di riserva per i file che non vengono caricati. Le parti dei file per quei set di lettura vengono trasferite nella posizione di fallback. Le posizioni di riserva sono disponibili per i sequence store creati dopo il 15 maggio 2023.

Crea una policy per i bucket di Amazon S3 per concedere l'accesso in HealthOmics scrittura alla posizione di fallback di Amazon S3, come illustrato nell'esempio seguente:

```
{
    "Effect": "Allow",
    "Principal": {
        "Service": "omics.amazonaws.com"
    },
    "Action": "s3:PutObject",
    "Resource": "arn:aws:s3:::amzn-s3-demo-bucket/*"
}
```

Se il bucket Amazon S3 per i fallback o i log di accesso utilizza una chiave gestita dal cliente, aggiungi le seguenti autorizzazioni alla policy chiave:

```
 {
    "Sid": "Allow use of key",
    "Effect": "Allow",
    "Principal": {
        "Service": "omics.amazonaws.com"
    },
    "Action": [
        "kms:Decrypt",
        "kms:GenerateDataKey*"
    ],
    "Resource": "*"
}
```

Esportazione di set di HealthOmics lettura in un bucket Amazon S3

Puoi esportare i set di lettura come processo di esportazione in batch in un bucket Amazon S3. Per farlo, crea innanzitutto una policy IAM con accesso in scrittura al bucket, simile al seguente esempio di policy IAM.

JSON

```
{
  "Version":"2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "s3:PutObject",
        "s3:GetBucketLocation"
      ],
      "Resource": [
        "arn:aws:s3:::amzn-s3-demo-bucket1",
        "arn:aws:s3:::amzn-s3-demo-bucket1/*"
      ]
    }
  ]
}
```

JSON

```
{
"Version":"2012-10-17",
"Statement": [
  {
      "Effect": "Allow",
      "Principal": {
          "Service": [
              "omics.amazonaws.com"
          ]
      },
      "Action": "sts:AssumeRole"
  }
]
}
```

Una volta stabilita la policy IAM, inizia il processo di esportazione read set. L'esempio seguente mostra come eseguire questa operazione utilizzando l'operazione API start-read-set-export-job.

Nell'esempio seguente, sostituisci tutti i parametri, come*sequence store ID*,*destination*, e *role ARNsources*, con il tuo input.

```
aws omics start-read-set-export-job
--sequence-store-id sequence store id \
--destination valid s3 uri \
--role-arn role ARN \
--sources readSetId=read set id_1 readSetId=read set id_2
```

Riceverai la seguente risposta con informazioni sull'archivio della sequenza di origine e sul bucket Amazon S3 di destinazione.

```
{
"id": <job-id>,
"sequenceStoreId": <sequence-store-id>,
"destination": <destination-s3-uri>,
"status": "SUBMITTED",
"creationTime": "2022-10-22T01:33:38.079000+00:00"
}
```

Dopo l'avvio del processo, puoi determinarne lo stato utilizzando l'operazione API get-read-set-export-job, come illustrato di seguito. Sostituisci *sequence store ID* e *job ID* con il tuo Sequence Store ID e Job ID, rispettivamente.

```
aws omics get-read-set-export-job --id job-id --sequence-store-id sequence store ID
```

È possibile visualizzare tutti i lavori di esportazione inizializzati per un archivio di sequenze utilizzando l'operazione API list-read-set-export-jobs, come illustrato di seguito. Sostituisci il *sequence store ID* con il tuo Sequence Store ID.

```
aws omics list-read-set-export-jobs --sequence-store-id sequence store ID.
```

```
{
"exportJobs": [
  {
      "id": <job-id>,
      "sequenceStoreId": <sequence-store-id>,
      "destination": <destination-s3-uri>,
      "status": "COMPLETED",
      "creationTime": "2022-10-22T01:33:38.079000+00:00",
```

```
      "completionTime": "2022-10-22T01:34:28.941000+00:00"
  }
]
}
```

Oltre a esportare i set di lettura, puoi anche condividerli utilizzando l'accesso Amazon URIs S3. Per ulteriori informazioni, consulta Accesso ai set di HealthOmics lettura con Amazon S3 URIs.

Accesso ai set di HealthOmics lettura con Amazon S3 URIs

Puoi utilizzare i percorsi URI di Amazon S3 per accedere ai set di lettura del tuo archivio di sequenze attivo.

Con il percorso URI di Amazon S3, puoi utilizzare le operazioni di Amazon S3 per elencare, condividere e scaricare i tuoi set di lettura. L'accesso tramite S3 APIs accelera la collaborazione e l'integrazione degli strumenti, dato che molti strumenti del settore sono già progettati per essere letti da S3. Inoltre, puoi condividere l'accesso a S3 APIs con altri account e fornire l'accesso in lettura ai dati in più regioni.

HealthOmics non supporta l'accesso URI di Amazon S3 ai set di lettura archiviati. Quando attivi un set di lettura, viene ripristinato ogni volta sullo stesso percorso URI.

Con i dati caricati negli HealthOmics store, poiché l'URI di Amazon S3 è basato sui punti di accesso Amazon S3, puoi integrarti direttamente con strumenti standard del settore che leggono Amazon S3, come i URIs seguenti:

- Applicazioni di analisi visiva come Integrative Genomics Viewer (IGV) o UCSC Genome Browser.
- Flussi di lavoro comuni con estensioni Amazon S3 come CWL, WDL e Nextflow.
- Qualsiasi strumento in grado di autenticare e leggere dal punto di accesso Amazon URIs S3 o leggere Amazon S3 prefirmato. URIs
- Utilità Amazon S3 come Mountpoint o. CloudFront

Amazon S3 Mountpoint consente di utilizzare un bucket Amazon S3 come file system locale. Per ulteriori informazioni su Mountpoint e per installarlo per l'uso, consulta Mountpoint per Amazon S3.

Amazon CloudFront è un servizio di rete per la distribuzione di contenuti (CDN) creato per prestazioni elevate, sicurezza e praticità per gli sviluppatori. Per ulteriori informazioni sull'uso di Amazon CloudFront, consulta la CloudFront documentazione di Amazon. Per configurare CloudFront un Sequence Store, contatta il AWS HealthOmics team.

L'account root del proprietario dei dati è abilitato per le azioni S3:GetObject, S3: e S3:List Bucket sul prefisso del Sequence Store. GetObjectTagging Per consentire a un utente dell'account di accedere ai dati, devi creare una policy IAM e collegarla all'utente o al ruolo. Per un esempio di policy, consulta Autorizzazioni per l'accesso ai dati tramite Amazon S3 URIs.

Puoi utilizzare le seguenti operazioni API di Amazon S3 sui set di lettura attivi per elencare e recuperare i tuoi dati. Puoi accedere ai set di lettura archiviati tramite Amazon URIs S3 dopo che sono stati attivati.

- GetObject— Recupera un oggetto da Amazon S3.
- HeadObject— L'operazione HEAD recupera i metadati da un oggetto senza restituire l'oggetto stesso. Questa operazione è utile se desiderate solo i metadati di un oggetto.
- ListObjects e ListObject v2 — Restituisce alcuni o tutti (fino a 1.000) gli oggetti in un bucket.
- CopyObject— Crea una copia di un oggetto già archiviato in Amazon S3. HealthOmicssupporta la copia su un punto di accesso Amazon S3, ma non la scrittura su un punto di accesso.

HealthOmics gli archivi di sequenze mantengono l'identità semantica dei file tramite. ETags Nel corso del ciclo di vita di un file, Amazon ETag S3, che si basa sull'identità bit per bit, può cambiare, HealthOmics ETag ma rimane lo stesso. Per ulteriori informazioni, consulta HealthOmics ETags e provenienza dei dati.

Argomenti

Struttura URI Amazon S3 nello storage HealthOmics

Tutti i file con Amazon S3 URIs dispongono di tag `omics:subjectId` di `omics:sampleId` risorsa. Puoi utilizzare questi tag per condividere l'accesso utilizzando le policy IAM attraverso un modello come`"s3:ExistingObjectTag/omics:subjectId": "pattern desired"`.

La struttura del file è la seguente:

`.../account_id/sequenceStore/seq_store_id/readSet/read_set_id/files`.

Per i file importati negli archivi di sequenza da Amazon S3, l'archivio di sequenze tenta di mantenere il nome sorgente originale. Quando i nomi sono in conflitto, il sistema aggiunge le informazioni sui set di lettura per garantire che i nomi dei file siano univoci. Ad esempio, per i set di lettura fastq, se entrambi i nomi di file sono uguali, per renderli unici, `sourceX` viene inserito prima di .fastq.gz o .fq.gz. Per un caricamento diretto, i nomi dei file seguono i seguenti schemi:

- Per FASTQ— *read_set_name* _ .fastq.gz *sourcex*
- uBAM/BAM/CRAMPer *read_set_name* —. *file extension*con estensioni di `.bam` o `.cram`. Un esempio è `NA193948.bam`.

Per i set di lettura che sono BAM o CRAM, i file di indice vengono generati automaticamente durante il processo di ingestione. Per i file di indice generati, viene applicata l'estensione di indice corretta alla fine del nome del file. Ha lo schema *<name of the Source the index is on>.<file index extension>.* Le estensioni dell'indice sono `.bai` o `.crai`.

Utilizzo di IGV ospitato o locale per accedere ai set di lettura

IGV è un browser genomico utilizzato per analizzare i file BAM e CRAM. Richiede sia il file che l'indice perché mostra solo una parte del genoma alla volta. IGV può essere scaricato e utilizzato localmente e sono disponibili guide per creare un IGV ospitato in AWS. La versione web pubblica non è supportata perché richiede CORS.

IGV locale si basa sulla AWS configurazione locale per accedere ai file. Assicurati che al ruolo utilizzato in quella configurazione sia associata una policy che kms: abiliti Decrypt e s3: GetObject autorizzazioni all'URI s3 dei set di lettura a cui si accede. Dopodiché, in IGV, puoi usare «File > carica da URL» e incollare l'URI per il codice sorgente e l'indice. In alternativa, presigned URLs può essere generato e utilizzato nello stesso modo, ignorando la configurazione AWS. Tieni presente che CORS non è supportato con l'accesso URI di Amazon S3, quindi le richieste che si basano su CORS non sono supportate.

L'esempio AWS Hosted IGV si affida ad AWS Cognito per creare le configurazioni e le autorizzazioni corrette all'interno dell'ambiente. Assicurati che venga creata una policy che abiliti le autorizzazioni KMS:Decrypt e s3: GetObject per l'URI Amazon S3 dei set di lettura a cui si accede e aggiungi questa policy al ruolo assegnato al pool di utenti Cognito. Dopodiché, in IGV, puoi usare «File > carica da URL» e inserire l'URI per l'origine e l'indice. In alternativa, presigned URLs può essere generato e utilizzato nello stesso modo, ignorando la configurazione AWS.

Tieni presente che l'archivio delle sequenze non verrà visualizzato nella scheda «Amazon» perché mostra solo i bucket di tua proprietà nella regione in cui è configurato il AWS profilo.

Utilizzando Samtools o in HTSlib HealthOmics

HTSlib è la libreria principale condivisa da diversi strumenti come Samtools, RSAMTools e altri. PySam Usa HTSlib la versione 1.20 o successiva per ottenere un supporto senza interruzioni per Amazon S3 Access Point. Per le versioni precedenti della HTSlib libreria, puoi utilizzare le seguenti soluzioni alternative:

- Imposta la variabile di ambiente per l'host HTS Amazon S3 con:. `export HTS_S3_HOST="s3.region.amazonaws.com"`
- Genera un URL predefinito per i file che desideri utilizzare. Se utilizzi un BAM o un CRAM, assicurati che venga generato un URL predefinito sia per il file che per l'indice. Dopodiché, entrambi i file possono essere utilizzati con le librerie.
- Usa Mountpoint per montare l'archivio di sequenze o leggere il prefisso set nello stesso ambiente in cui stai usando HTSlib le librerie. Da qui, è possibile accedere ai file utilizzando i percorsi dei file locali.

Usare Mountpoint HealthOmics

Mountpoint per Amazon S3 è un client di file semplice e ad alta velocità per il montaggio di un bucket Amazon S3 come file system locale. Con Mountpoint per Amazon S3, le tue applicazioni possono accedere agli oggetti archiviati in Amazon S3 tramite operazioni sui file come apertura e lettura. Mountpoint per Amazon S3 traduce automaticamente queste operazioni in chiamate API di oggetti Amazon S3, offrendo alle applicazioni l'accesso allo storage elastico e al throughput di Amazon S3 tramite un'interfaccia di file.

Mountpoint può essere installato utilizzando le istruzioni di installazione di Mountpoint. Mountpoint utilizza il profilo AWS locale per l'installazione e funziona a livello di prefisso Amazon S3. Assicurati che il profilo utilizzato abbia una policy che abiliti le autorizzazioni s3:GetObject, s3: ListBucket e kms: Decrypt per il prefisso URI Amazon S3 dei set di lettura o dell'archivio di sequenze a cui si accede. Successivamente, il bucket può essere montato utilizzando il seguente percorso:

```
mount-s3 access point arn local path to mount --prefix prefix to sequence store or read
 set --region region
```

Usando con CloudFront HealthOmics

Amazon CloudFront è un servizio di rete per la distribuzione di contenuti (CDN) progettato per prestazioni elevate, sicurezza e praticità per gli sviluppatori. I clienti che lo desiderano CloudFront devono collaborare con il team di assistenza per attivare la CloudFront distribuzione. Collabora con il team del tuo account per coinvolgere il team HealthOmics di assistenza.

Attivazione dei set di lettura in HealthOmics

È possibile attivare i set di lettura archiviati con l'operazione API start-read-set-activation-job o tramite AWS CLI, come illustrato nell'esempio seguente. Sostituisci *sequence store ID* e *read set id* con il tuo sequence store ID e read set. IDs

```
aws omics start-read-set-activation-job
     --sequence-store-id sequence store ID \
     --sources readSetId=read set ID readSetId=read set id_1 read set id_2
```

Riceverai una risposta che contiene le informazioni sul processo di attivazione, come illustrato di seguito.

```
{
    "id": "12345678",
    "sequenceStoreId": "1234567890",
    "status": "SUBMITTED",
    "creationTime": "2022-10-22T00:50:54.670000+00:00"
}
```

Dopo l'avvio del processo di attivazione, puoi monitorarne l'avanzamento con l'operazione API get-read-set-activation-job. Di seguito è riportato un esempio di come utilizzare il per AWS CLI verificare lo stato del processo di attivazione. Sostituisci *job ID* e *sequence store ID* con il tuo Sequence Store ID e Job IDs, rispettivamente.

```
aws omics get-read-set-activation-job --id job ID --sequence-store-id sequence store ID
```

La risposta riassume il processo di attivazione, come illustrato di seguito.

```
{
```

```
    "id": 123567890,
    "sequenceStoreId": 123467890,
    "status": "SUBMITTED",
    "statusUpdateReason": "The job is submitted and will start soon.",
    "creationTime": "2022-10-22T00:50:54.670000+00:00",
    "sources": [
        {
            "readSetId": <reads set id_1>,
            "status": "NOT_STARTED",
            "statusUpdateReason": "The source is queued for the job."
        },
        {
            "readSetId": <read set id_2>,
            "status": "NOT_STARTED",
            "statusUpdateReason": "The source is queued for the job."
        }
    ]
}
```

È possibile verificare lo stato di un processo di attivazione tramite l'operazione get-read-set-metadataAPI. Gli stati possibili sono `ACTIVEACTIVATING`, e`ARCHIVED`. Nell'esempio seguente, sostituitelo *`sequence store ID`* con il vostro Sequence Store ID e sostituitelo *`read set ID`* con il vostro Read Set ID.

```
aws omics get-read-set-metadata --sequence-store-id sequence store ID --id read set ID
```

La risposta seguente mostra che il set di lettura è attivo.

```
{
    "id": "12345678",
    "arn": "arn:aws:omics:us-west-2:555555555555:sequenceStore/1234567890/
readSet/12345678",
    "sequenceStoreId": "0123456789",
    "subjectId": "mySubject",
    "sampleId": "mySample",
    "status": "ACTIVE",
    "name": "HG00100",
    "description": "HG00100 aligned to HG38 BAM",
    "fileType": "BAM",
    "creationTime": "2022-07-13T23:25:20Z",
    "sequenceInformation": {
        "totalReadCount": 1513467,
```

```
        "totalBaseCount": 163454436,
        "generatedFrom": "Pulled from SRA",
        "alignment": "ALIGNED"
    },
    "referenceArn": "arn:aws:omics:us-west-2:555555555555:referenceStore/0123456789/
reference/0000000001",
    "files": {
        "source1": {
            "totalParts": 2,
            "partSize":  10485760,
            "contentLength": 17112283,
            "s3Access": {
        "s3Uri": "s3://accountID-sequence store ID-ajdpi90jdas90a79fh9a8ja98jdfa9jf98-
s3alias/592761533288/sequenceStore/2015356892/readSet/9515444019/
import_source1.fastq.gz"
},
         },
        "index": {
            "totalParts": 1,
            "partSize": 53216,
            "contentLength": 10485760
            "s3Access": {
        "s3Uri": "s3://accountID-sequence store ID-ajdpi90jdas90a79fh9a8ja98jdfa9jf98-
s3alias/592761533288/sequenceStore/2015356892/readSet/9515444019/
import_source1.fastq.gz"
},
        }
    },
    "creationType": "IMPORT",
    "etag": {
        "algorithm": "BAM_MD5up",
        "source1": "d1d65429212d61d115bb19f510d4bd02"
    }
}
```


È possibile visualizzare tutti i processi di attivazione del set di lettura utilizzando list-read-set-activation-jobs, come mostrato nell'esempio seguente. Nell'esempio seguente, sostituiscilo *`sequence store ID`* con il tuo Sequence Store ID.

```
aws omics list-read-set-activation-jobs --sequence-store-id sequence store ID
```

Riceverai la seguente risposta.

```
{
    "activationJobs": [
        {
            "id": 1234657890,
            "sequenceStoreId": "1234567890",
            "status": "COMPLETED",
            "creationTime": "2022-10-22T01:33:38.079000+00:00",
            "completionTime": "2022-10-22T01:34:28.941000+00:00"
        }
    ]
}
```

HealthOmics analisi

> **Important**
>
> AWS HealthOmics i negozi di varianti e i negozi di annotazioni non sono più aperti a nuovi clienti. I clienti esistenti possono continuare a utilizzare il servizio normalmente. Per ulteriori informazioni, consulta AWS HealthOmics modifica della disponibilità dell'archivio delle varianti e dell'archivio delle annotazioni.

HealthOmics l'analisi supporta l'archiviazione e l'analisi di varianti e annotazioni genomiche. Analytics fornisce due tipi di risorse di archiviazione: gli archivi di varianti e gli archivi di annotazioni. Queste risorse vengono utilizzate per archiviare, trasformare e interrogare i dati delle varianti genomiche e i dati di annotazione. Dopo aver importato i dati in un datastore, puoi utilizzare Athena per eseguire analisi avanzate sui dati.

Puoi utilizzare la HealthOmics console o l'API per creare e gestire archivi, importare dati e condividere i dati degli archivi analitici con i collaboratori.

Gli archivi Variant supportano i dati nei formati VCF e gli archivi di annotazione supportano il supporto e i formati. TSV/CSV GFF3 Le coordinate genomiche sono rappresentate come intervalli a base zero, semichiusi e semiaperti. Quando i dati si trovano nell'archivio dati di HealthOmics analisi, l'accesso ai file VCF viene gestito tramite. AWS Lake Formation Puoi quindi interrogare i file VCF utilizzando Amazon Athena. Le interrogazioni devono utilizzare il motore di interrogazione Athena versione 3. Per ulteriori informazioni sulle versioni del motore di query Athena, consulta la documentazione di Amazon Athena.

Argomenti

- Interrogazione dei HealthOmics dati di analisi
- Condivisione di archivi HealthOmics di analisi

Creazione di negozi di HealthOmics varianti

> **Important**
>
> AWS HealthOmics i negozi di varianti e i negozi di annotazioni non sono più aperti a nuovi clienti. I clienti esistenti possono continuare a utilizzare il servizio normalmente. Per ulteriori informazioni, consulta AWS HealthOmics modifica della disponibilità dell'archivio delle varianti e dell'archivio delle annotazioni.

I seguenti argomenti descrivono come creare negozi di HealthOmics varianti utilizzando la console e l'API.

Argomenti

- Creazione di un archivio di varianti utilizzando la console
- Creazione di un negozio di varianti utilizzando l'API

Creazione di un archivio di varianti utilizzando la console

Puoi creare un negozio di varianti utilizzando la HealthOmics console.

1. Apri la HealthOmics console.
2. Se necessario, apri il riquadro di navigazione a sinistra (≡). Scegli Variant stores.
3. Nella pagina Crea negozio di varianti, fornisci le seguenti informazioni
   - Nome del negozio di varianti: un nome univoco per questo negozio.
   - Descrizione (opzionale): una descrizione di questo negozio di varianti.
   - Genoma di riferimento: il genoma di riferimento per questo archivio di varianti.
   - Crittografia dei dati: scegli se desideri che la crittografia dei dati sia di proprietà e gestita da te AWS o da solo.
   - Tag (opzionale): fornisci fino a 50 tag per questo negozio di varianti.
4. Scegli Crea negozio di varianti.

Creazione di un negozio di varianti utilizzando l'API

Utilizza l'operazione HealthOmics `CreateVariantStore` API per creare archivi di varianti. È possibile eseguire questa operazione anche con AWS CLI.

Per creare un negozio di varianti, fornisci un nome per il negozio e l'ARN di un negozio di riferimento. L'archivio delle varianti è pronto per l'inserimento dei dati quando il relativo stato passa a READY.

L'esempio seguente utilizza il AWS CLI per creare un negozio di varianti.

```
aws omics create-variant-store --name myvariantstore \
    --reference referenceArn="arn:aws:omics:us-
west-2:555555555555:referenceStore/123456789/reference/5987565360"
```

Per confermare la creazione del tuo negozio di varianti, riceverai la seguente risposta.

```
{
    "creationTime": "2022-11-03T18:19:52.296368+00:00",
    "id": "45aeb91d5678",
    "name": "myvariantstore",
    "reference": {
        "referenceArn": "arn:aws:omics:us-west-2:555555555555:referenceStore/123456789/
reference/5987565360"
    },
    "status": "CREATING"
}
```

Per saperne di più su un negozio di varianti, utilizza l'get-variant-storeAPI.

```
aws omics get-variant-store --name myvariantstore
```

Riceverai la seguente risposta.

```
{
    "id": "45aeb91d5678",
    "reference": {
        "referenceArn": "arn:aws:omics:us-west-2:555555555555:referenceStore/123456789/
reference/5987565360"
    },
    "status": "ACTIVE",
    "storeArn": "arn:aws:omics:us-west-2:555555555555:variantStore/myvariantstore",
```

```
    "name": "myvariantstore",
    "creationTime": "2022-11-03T18:19:52.296368+00:00",
    "updateTime": "2022-11-03T18:30:56.272792+00:00",
    "tags": {},
    "storeSizeBytes": 0
}
```

Per visualizzare tutti i negozi di varianti associati a un account, utilizza l'list-variant-storesAPI.

```
aws omics list-variant-stores
```

Riceverai una risposta che elenca tutti gli store di varianti IDs, insieme ai relativi stati e ad altri dettagli, come mostrato nella risposta di esempio seguente.

```
{
    "variantStores": [
        {
            "id": "45aeb91d5678",
            "reference": {
                "referenceArn": "arn:aws:omics:us-
west-2:55555555555:referenceStore/5506874698"
            },
            "status": "ACTIVE",
            "storeArn": "arn:aws:omics:us-west-2:55555555555:variantStore/
new_variant_store",
            "name": "variantstore",
            "creationTime": "2022-11-03T18:19:52.296368+00:00",
            "updateTime": "2022-11-03T18:30:56.272792+00:00",
            "statusMessage": "",
            "storeSizeBytes": 141526
        }
    ]
}
```

Puoi anche filtrare le risposte per l'list-variant-storesAPI in base agli stati o ad altri criteri.

I file VCF importati negli archivi analitici creati il o dopo il 15 maggio 2023 hanno schemi definiti per le annotazioni Variant Effect Predictor (VEP). Ciò semplifica l'interrogazione e l'analisi dei dati VCF importati. La modifica non ha alcun impatto sui negozi creati prima del 15 maggio 2023, a meno che il `annotation fields` parametro non sia incluso nella chiamata API o CLI. Per questi negozi, l'utilizzo del `annotation fields` parametro causerà l'esito negativo della richiesta.

Creazione di processi di importazione di HealthOmics varianti di store

> **Important**
>
> AWS HealthOmics i negozi di varianti e gli archivi di annotazioni non sono più aperti a nuovi clienti. I clienti esistenti possono continuare a utilizzare il servizio normalmente. Per ulteriori informazioni, consulta AWS HealthOmics modifica della disponibilità dell'archivio delle varianti e dell'archivio delle annotazioni.

L'esempio seguente mostra come utilizzare per AWS CLI creare un processo di importazione per un negozio di varianti.

```
aws omics start-variant-import-job \
       --destination-name myvariantstore \
       --runLeftNormalization false \
       --role-arn  arn:aws:iam::55555555555:role/roleName \
       --items source=s3://my-omics-bucket/sample.vcf.gz source=s3://my-omics-bucket/
sample2.vcf.gz
```

```
{
    "destinationName": "store_a",
    "roleArn": "....",
    "runLeftNormalization": false,
    "items": [
        {"source": "s3://my-omics-bucket/sample.vcf.gz"},
        {"source": "s3://my-omics-bucket/sample2.vcf.gz"}
    ]
}
```

Per i negozi creati dopo il 15 maggio 2023, l'esempio seguente mostra come aggiungere il `--annotation-fields` parametro. I campi di annotazione vengono definiti con l'importazione.

```
aws omics start-variant-import-job \
   --destination-name annotationparsingvariantstore \
   --role-arn arn:aws:iam::123456789012:role/<role_name> \
   --items source=s3://pathToS3/sample.vcf
   --annotation-fields '{"VEP": "CSQ"}'
```

```
{
    "jobId": "981e2286-e954-4391-8a97-09aefc343861"
}
```

get-variant-import-jobUsare per controllare lo stato.

```
aws omics get-variant-import-job --job-id 08279950-a9e3-4cc3-9a3c-a574f9c9e229
```

Riceverai una risposta JSON che mostra lo stato del tuo processo di importazione. Le annotazioni VEP nel VCF vengono analizzate alla ricerca di informazioni memorizzate nella colonna INFO in coppia. ID/Value L'ID predefinito per la colonna INFO delle annotazioni di Ensembl Variant Effect Predictor è CSQ, ma è possibile utilizzare il `--annotation-fields` parametro per indicare un valore personalizzato utilizzato nella colonna INFO. L'analisi è attualmente supportata per le annotazioni VEP.

Per un negozio creato prima del 15 maggio 2023 o per i file VCF che non includono l'annotazione VEP, la risposta non include alcun campo di annotazione.

```
{
    "creationTime": "2023-04-11T17:52:37.241958+00:00",
    "destinationName": "annotationparsingvariantstore",
    "id": "7a1c67e3-b7f9-434d-817b-9c571fd63bea",
    "items": [

    {
       "jobStatus": "COMPLETED",
       "source": "s3://amzn-s3-demo-bucket/NA12878.2k.garvan.vcf"
    }
 ],
    "roleArn": "arn:aws:iam::555555555555:role/<role_name>",

    "runLeftNormalization": false,
    "status": "COMPLETED",
    "updateTime": "2023-04-11T17:58:22.676043+00:00",
}
```

Le annotazioni VEP che fanno parte dei file VCF vengono archiviate come schema predefinito con la seguente struttura. Il campo extras può essere utilizzato per memorizzare eventuali campi VEP aggiuntivi non inclusi nello schema predefinito.

```
annotations struct<
   vep: array<struct<
      allele:string,
      consequence: array<string>,
      impact:string,
      symbol:string,
      gene:string,
      `feature_type`: string,
      feature: string,
      biotype: string,
      exon: struct<rank:string, total:string>,
      intron: struct<rank:string, total:string>,
      hgvsc: string,
      hgvsp: string,
      `cdna_position`: string,
      `cds_position`: string,
      `protein_position`: string,
      `amino_acids`: struct<reference:string, variant: string>,
      codons: struct<reference:string, variant: string>,
      `existing_variation`: array<string>,
      distance: string,
      strand: string,
      flags: array<string>,
      symbol_source: string,
      hgnc_id: string,
      `extras`: map<string, string>
    >>
>
```

L'analisi viene eseguita con il massimo impegno. Se la voce VEP non segue le specifiche dello standard VEP, non verrà analizzata e la riga dell'array sarà vuota.

Per un nuovo archivio di varianti, la risposta per get-variant-import-jobincluderebbe i campi di annotazione, come mostrato.

```
aws omics get-variant-import-job --job-id 08279950-a9e3-4cc3-9a3c-a574f9c9e229
```

Riceverai una risposta JSON che mostra lo stato del tuo processo di importazione.

```
{
    "creationTime": "2023-04-11T17:52:37.241958+00:00",
```

```
    "destinationName": "annotationparsingvariantstore",
    "id": "7a1c67e3-b7f9-434d-817b-9c571fd63bea",
    "items": [

    {
    "jobStatus": "COMPLETED",
    "source": "s3://amzn-s3-demo-bucket/NA12878.2k.garvan.vcf"
    }
 ],
    "roleArn": "arn:aws:iam::123456789012:role/<role_name>",
    "runLeftNormalization": false,
    "status": "COMPLETED",
    "updateTime": "2023-04-11T17:58:22.676043+00:00",
    "annotationFields" : {"VEP": "CSQ"}
  }
}
```

Puoi usarla list-variant-import-jobsper vedere tutti i lavori di importazione e i relativi stati.

```
aws omics list-variant-import-jobs --ids 7a1c67e3-b7f9-434d-817b-9c571fd63bea
```

La risposta contiene le seguenti informazioni.

```
{
    "variantImportJobs": [
    {
        "creationTime": "2023-04-11T17:52:37.241958+00:00",
        "destinationName": "annotationparsingvariantstore",
        "id": "7a1c67e3-b7f9-434d-817b-9c571fd63bea",
        "roleArn": "arn:aws:iam::55555555555:role/roleName",
        "runLeftNormalization": false,
        "status": "COMPLETED",
        "updateTime": "2023-04-11T17:58:22.676043+00:00",
        "annotationFields" : {"VEP": "CSQ"}
        }
    ]
  }
}
```

Se necessario, è possibile annullare un processo di importazione con il comando seguente.

```
aws omics cancel-variant-import-job
```

```
    --job-id edd7b8ce-xmpl-47e2-bc99-258cac95a508
```

Creazione di HealthOmics archivi di annotazioni

> **Important**
>
> AWS HealthOmics i negozi di varianti e i negozi di annotazioni non sono più aperti a nuovi clienti. I clienti esistenti possono continuare a utilizzare il servizio normalmente. Per ulteriori informazioni, consulta AWS HealthOmics modifica della disponibilità dell'archivio delle varianti e dell'archivio delle annotazioni.

Un archivio di annotazioni è un archivio dati che rappresenta un database di annotazioni, ad esempio uno proveniente da un file TSV, VCF o GFF. Se viene specificato lo stesso genoma di riferimento, gli archivi di annotazioni vengono mappati sullo stesso sistema di coordinate degli archivi di varianti durante l'importazione. I seguenti argomenti mostrano come utilizzare la HealthOmics console e creare e AWS CLI gestire archivi di annotazioni.

Argomenti

Creazione di un archivio di annotazioni utilizzando la console

Utilizzare la procedura seguente per creare archivi di annotazioni con la HealthOmics console.

Per creare un archivio di annotazioni

1. Apri la HealthOmics console.
2. Se necessario, apri il riquadro di navigazione a sinistra (≡). Scegliete Annotation stores.
3. Nella pagina Archivi di annotazioni, scegli Crea archivio di annotazioni.
4. Nella pagina Crea archivio di annotazioni, fornisci le seguenti informazioni
   - Nome dell'archivio di annotazioni: un nome univoco per questo negozio.
   - Descrizione (opzionale): una descrizione di questo genoma di riferimento.

- Dettagli sul formato dei dati e sullo schema: seleziona il formato del file di dati e carica la definizione dello schema per questo archivio.
- Genoma di riferimento: il genoma di riferimento per questa annotazione.
- Crittografia dei dati: scegli se desideri che la crittografia dei dati sia di proprietà e gestita da te AWS o da solo.
- Tag (opzionale): fornisci fino a 50 tag per questo archivio di annotazioni.

5. Scegli Crea archivio di annotazioni.

Creazione di un archivio di annotazioni utilizzando l'API

L'esempio seguente mostra come creare un archivio di annotazioni utilizzando. AWS CLI Per tutte le operazioni AWS CLI e le API, è necessario specificare il formato dei dati.

```
aws omics create-annotation-store --name my_annotation_store \
           --store-format GFF \
           --reference referenceArn="arn:aws:omics:us-
west-2:555555555555:referenceStore/6505293348/reference/5987565360"
           --version-name new_version
```

Riceverai la seguente risposta per confermare la creazione del tuo archivio di annotazioni.

```
{
           "creationTime": "2022-08-24T20:34:19.229500Z",
           "id": "3b93cdef69d2",
           "name": "my_annotation_store",
           "reference": {
               "referenceArn": "arn:aws:omics:us-
west-2:555555555555:referenceStore/6505293348/reference/5987565360"
           },
           "status": "CREATING"
           "versionName": "my_version"
       }
```

Per saperne di più su un archivio di annotazioni, utilizza l'get-annotation-storeAPI.

```
aws omics get-annotation-store --name my_annotation_store
```

Riceverai la seguente risposta.

```
{
          "id": "eeb019ac79c2",
          "reference": {
              "referenceArn": "arn:aws:omics:us-
west-2:555555555555:referenceStore/5638433913/reference/5871590330“
          },
          "status": "ACTIVE",
          "storeArn": "arn:aws:omics:us-west-2:555555555555:annotationStore/gffstore",
          "name": "my_annotation_store",
          "creationTime": "2022-11-05T00:05:19.136131+00:00",
          "updateTime": "2022-11-05T00:10:36.944839+00:00",
          "tags": {},
          "storeFormat": "GFF",
          "statusMessage": "",
          "storeSizeBytes": 0,
          "numVersions": 1
      }
```

Per visualizzare tutti gli archivi di annotazioni associati a un account, utilizza l'operazione list-annotation-storesAPI.

```
aws omics list-annotation-stores
```

Riceverai una risposta che elenca tutti gli archivi di annotazioni IDs, insieme ai relativi stati e ad altri dettagli, come mostrato nella risposta di esempio seguente.

```
{
           "annotationStores": [
               {
                  "id": "4d8f3eada259",
                   "reference":
                       "referenceArn": "arn:aws:omics:us-
west-2:555555555555:referenceStore/5638433913/reference/5871590330"
                   },
                   "status": "CREATING",
                   "name": "gffstore",
                   "creationTime": "2022-09-27T17:30:52.182990+00:00",
                   "updateTime": "2022-09-27T17:30:53.025362+00:00"
               }
           ]
       }
```

È inoltre possibile filtrare le risposte in base allo stato o ad altri criteri.

Creazione di processi di importazione per gli HealthOmics archivi di annotazioni

> **Important**
>
> AWS HealthOmics gli store di varianti e gli archivi di annotazioni non sono più aperti a nuovi clienti. I clienti esistenti possono continuare a utilizzare il servizio normalmente. Per ulteriori informazioni, consulta AWS HealthOmics modifica della disponibilità dell'archivio delle varianti e dell'archivio delle annotazioni.

Argomenti

Creazione di un processo di importazione delle annotazioni utilizzando l'API

L'esempio seguente mostra come utilizzare per avviare un AWS CLI processo di importazione di annotazioni.

```
aws omics start-annotation-import-job \
           --destination-name myannostore \
           --version-name myannostore \
           --role-arn arn:aws:iam::123456789012:role/roleName \
           --items source=s3://my-omics-bucket/sample.vcf.gz
           --annotation-fields '{"VEP": "CSQ"}'
```

Gli archivi di annotazioni creati prima del 15 maggio 2023 restituiscono un messaggio di errore se i campi di annotazione sono inclusi. Non restituiscono l'output per le operazioni API coinvolte nei processi di importazione di annotation store.

È quindi possibile utilizzare l'operazione get-annotation-import-jobAPI e il `job ID` parametro per ottenere maggiori dettagli sul processo di importazione delle annotazioni.

```
aws omics get-annotation-import-job --job-id 9e4198fb-fa85-446c-9301-9b823a1a8ba8
```

Riceverai la seguente risposta, inclusi i campi di annotazione.

```
{
          "creationTime": "2023-04-11T19:09:25.049767+00:00",
          "destinationName": "parsingannotationstore",
          "versionName": "parsingannotationstore",
          "id": "9e4198fb-fa85-446c-9301-9b823a1a8ba8",
          "items": [
              {
                  "jobStatus": "COMPLETED",
                  "source": "s3://my-omics-bucket/sample.vep.vcf"
              }
          ],
          "roleArn": "arn:aws:iam::55555555555:role/roleName",
          "runLeftNormalization": false,
          "status": "COMPLETED",
          "updateTime": "2023-04-11T19:13:09.110130+00:00",
          "annotationFields" : {"VEP": "CSQ"}
       }
```

Per visualizzare tutti i lavori di importazione di Annotation Store, utilizzare. list-annotation-import-jobs

```
aws omics list-annotation-import-jobs --ids 9e4198fb-fa85-446c-9301-9b823a1a8ba8
```

La risposta include i dettagli e lo stato dei lavori di importazione dell'archivio di annotazioni.

```
{
          "annotationImportJobs": [
          {
              "creationTime": "2023-04-11T19:09:25.049767+00:00",
              "destinationName": "parsingannotationstore",
              "versionName": "parsingannotationstore",
              "id": "9e4198fb-fa85-446c-9301-9b823a1a8ba8",
              "roleArn": "arn:aws:iam::55555555555:role/roleName",
              "runLeftNormalization": false,
              "status": "COMPLETED",
```

```
            "updateTime": "2023-04-11T19:13:09.110130+00:00",
            "annotationFields" : {"VEP": "CSQ"}
        }
        ]
    }
```

Parametri aggiuntivi per i formati TSV e VCF

Per i formati TSV e VCF, esistono parametri aggiuntivi che informano l'API su come analizzare l'input.

> **Important**
>
> I dati di annotazione CSV esportati con i motori di query restituiscono direttamente le informazioni dall'importazione del set di dati. Se i dati importati contengono formule o comandi, il file potrebbe essere soggetto all'iniezione di file CSV. Pertanto, i file esportati con i motori di query possono richiedere avvisi di sicurezza. Per evitare attività dannose, disattivate i link e le macro durante la lettura dei file di esportazione.

Il parser TSV esegue anche operazioni bioinformatiche di base, come la normalizzazione sinistra e la standardizzazione delle coordinate genomiche, elencate nella tabella seguente.

Tipo di formato	Description
Generico	File di testo generico. Nessuna informazione genomica.
`CHR_POS`	Posizione iniziale - 1, Aggiungi posizione finale, che è la stessa di. `POS`
`CHR_POS_REF_ALT`	Contiene informazioni sugli alleli contig, 1-base position, ref e alt.
`CHR_START_END_REF_ALT_ONE_BASE`	Contiene informazioni sugli alleli contig, start, end, ref e alt. Le coordinate sono a base 1.
`CHR_START_END_ZERO_BASE`	Contiene le posizioni contig, iniziale e finale. Le coordinate sono basate su 0.

Tipo di formato	Description
CHR_START_END_ONE_BASE	Contiene le posizioni contig, iniziale e finale. Le coordinate sono a base 1.
CHR_START_END_REF_ALT_ZERO_BASE	Contiene informazioni sugli alleli contig, start, end, ref e alt. Le coordinate sono basate su 0.

Una richiesta di archiviazione delle annotazioni di importazione TSV è simile all'esempio seguente.

```
aws omics start-annotation-import-job \
--destination-name tsv_anno_example \
--role-arn arn:aws:iam::555555555555:role/demoRole \
--items source=s3://demodata/genomic_data.bed.gz \
--format-options '{ "tsvOptions": {
        "readOptions": {
            "header": false,
            "sep": "\t"
        }
    }
}'
```

Creazione di archivi di annotazioni in formato TSV

L'esempio seguente crea un archivio di annotazioni utilizzando un file con limiti di tabulazioni che contiene un'intestazione, righe e commenti. Le coordinate sono CHR_START_END_ONE_BASED e contiene la mappa HG19 genica della Synopsis of the Human Gene Map dell'OMIM.

```
aws omics create-annotation-store --name mimgenemap \
  --store-format TSV \
  --reference=referenceArn=arn:aws:omics:us-
west-2:555555555555:referenceStore/6505293348/reference/2310864158 \
  --store-options=tsvStoreOptions='{
    annotationType=CHR_START_END_ONE_BASE,
    formatToHeader={CHR=chromosome, START=genomic_position_start,
 END=genomic_position_end},
    schema=[
      {chromosome=STRING},
      {genomic_position_start=LONG},
```

```
      {genomic_position_end=LONG},
      {cyto_location=STRING},
      {computed_cyto_location=STRING},
      {mim_number=STRING},
      {gene_symbols=STRING},
      {gene_name=STRING},
      {approved_gene_name=STRING},
      {entrez_gene_id=STRING},
      {ensembl_gene_id=STRING},
      {comments=STRING},
      {phenotypes=STRING},
      {mouse_gene_symbol=STRING}]}'
```

È possibile importare file con o senza un'intestazione. Per indicarlo in una richiesta CLI, utilizzare`header=false`, come mostrato nel seguente esempio di processo di importazione.

```
aws omics start-annotation-import-job \
   --role-arn arn:aws:iam::555555555555:role/demoRole \
   --items=source=s3://amzn-s3-demo-bucket/annotation-examples/hg38_genemap2.txt \
   --destination-name output-bucket \
   --format-options=tsvOptions='{readOptions={sep="\t",header=false,comment="#"}}'
```

L'esempio seguente crea un archivio di annotazioni per un file bed. Un file bed è un semplice file delimitato da tabulazioni. In questo esempio, le colonne sono cromosoma, inizio, fine e nome della regione. Le coordinate sono a base zero e i dati non hanno un'intestazione.

```
aws omics create-annotation-store \
   --name cexbed --store-format TSV \
   --reference=referenceArn=arn:aws:omics:us-
west-2:555555555555:referenceStore/6505293348/reference/2310864158 \
   --store-options=tsvStoreOptions='{
   annotationType=CHR_START_END_ZERO_BASE,
   formatToHeader={CHR=chromosome, START=start, END=end},
   schema=[{chromosome=STRING}, {start=LONG}, {end=LONG}, {name=STRING}]}'
```

È quindi possibile importare il file bed nell'archivio di annotazioni utilizzando il seguente comando CLI.

```
aws omics start-annotation-import-job \
   --role-arn arn:aws:iam::555555555555:role/demoRole \
   --items=source=s3://amzn-s3-demo-bucket/TruSeq_Exome_TargetedRegions_v1.2.bed \
```

```
   --destination-name cexbed \
   --format-options=tsvOptions='{readOptions={sep="\t",header=false,comment="#"}}'
```

L'esempio seguente crea un archivio di annotazioni per un file delimitato da tabulazioni che contiene le prime colonne di un file VCF, seguite da colonne con informazioni sulle annotazioni. Contiene le posizioni del genoma con informazioni sul cromosoma, sugli alleli iniziali, di riferimento e alternativi e contiene un'intestazione.

```
aws omics create-annotation-store --name gnomadchrx --store-format TSV \
--reference=referenceArn=arn:aws:omics:us-
west-2:555555555555:referenceStore/6505293348/reference/2310864158 \
--store-options=tsvStoreOptions='{
    annotationType=CHR_POS_REF_ALT,
    formatToHeader={CHR=chromosome, POS=start, REF=ref, ALT=alt},
    schema=[
        {chromosome=STRING},
        {start=LONG},
        {ref=STRING},
        {alt=STRING},
        {filters=STRING},
        {ac_hom=STRING},
        {ac_het=STRING},
        {af_hom=STRING},
        {af_het=STRING},
        {an=STRING},
        {max_observed_heteroplasmy=STRING}]}'
```

È quindi necessario importare il file nell'archivio di annotazioni utilizzando il seguente comando CLI.

```
aws omics start-annotation-import-job \
  --role-arn arn:aws:iam::555555555555:role/demoRole \
   --items=source=s3://amzn-s3-demo-bucket/
gnomad.genomes.v3.1.sites.chrM.reduced_annotations.tsv \
   --destination-name gnomadchrx \
   --format-options=tsvOptions='{readOptions={sep="\t",header=true,comment="#"}}'
```

L'esempio seguente mostra come un cliente può creare un archivio di annotazioni per un file mim2gene. Un file mim2gene fornisce i collegamenti tra i geni in OMIM e un altro identificatore genico. È delimitato da tabulazioni e contiene commenti.

```
aws omics create-annotation-store \
  --name mim2gene \
  --store-format TSV \
  --reference=referenceArn=arn:aws:omics:us-
west-2:555555555555:referenceStore/6505293348/reference/2310864158 \
  --store-options=tsvStoreOptions='
    {annotationType=GENERIC,
    formatToHeader={},
    schema=[
        {mim_gene_id=STRING},
        {mim_type=STRING},
        {entrez_id=STRING},
        {hgnc=STRING},
        {ensembl=STRING}]}'
```

Puoi quindi importare i dati nel tuo negozio come segue.

```
aws omics start-annotation-import-job \
   --role-arn arn:aws:iam::555555555555:role/demoRole \
   --items=source=s3://xquek-dev-aws/annotation-examples/mim2gene.txt \
   --destination-name mim2gene \
   --format-options=tsvOptions='{readOptions={sep="\t",header=false,comment="#"}}'
```

Avvio di processi di importazione in formato VCF

Per i file VCF, sono disponibili due input aggiuntivi che ignorano o includono tali parametri come mostrato. `ignoreQualField` `ignoreFilterField`

```
aws omics start-annotation-import-job --destination-name annotation_example\
  --role-arn arn:aws:iam::555555555555:role/demoRole \
  --items source=s3://demodata/example.garvan.vcf \
  --format-options '{ "vcfOptions": {
    "ignoreQualField": false,
    "ignoreFilterField": false
    }
   }'
```

È inoltre possibile annullare l'importazione di un archivio di annotazioni, come illustrato. Se l'annullamento ha esito positivo, non riceverai una risposta a questa AWS CLI chiamata. Tuttavia, se

l'ID del processo di importazione non viene trovato o il processo di importazione è completato, viene visualizzato un messaggio di errore.

```
aws omics cancel-annotation-import-job --job-id edd7b8ce-xmpl-47e2-bc99-258cac95a508
```

> **Note**
>
> I metadati importano la cronologia dei lavori per get-annotation-import-job, get-variant-import-joblist-annotation-import-jobs, e list-variant-import-jobsvengono eliminati automaticamente dopo due anni. I dati di varianti e annotazioni importati non vengono eliminati automaticamente e rimangono nei tuoi archivi dati.

Creazione di HealthOmics versioni dell'Annotation Store

> **Important**
>
> AWS HealthOmics i negozi di varianti e i negozi di annotazioni non sono più aperti a nuovi clienti. I clienti esistenti possono continuare a utilizzare il servizio normalmente. Per ulteriori informazioni, consulta AWS HealthOmics modifica della disponibilità dell'archivio delle varianti e dell'archivio delle annotazioni.

È possibile creare nuove versioni degli archivi di annotazioni per raccogliere diverse versioni dei database di annotazioni. In questo modo è possibile organizzare i dati delle annotazioni, che vengono aggiornati regolarmente.

Per creare una nuova versione di un archivio di annotazioni esistente, utilizza l'create-annotation-store-versionAPI come mostrato nell'esempio seguente.

```
aws omics create-annotation-store-version \
     --name my_annotation_store \
     --version-name my_version
```

Riceverai la seguente risposta con l'ID della versione dell'Annotation Store, che conferma che è stata creata una nuova versione dell'annotazione.

```
{
```

```
     "creationTime": "2023-07-21T17:15:49.251040+00:00",
     "id": "3b93cdef69d2",
     "name": "my_annotation_store",
     "reference": {
         "referenceArn": "arn:aws:omics:us-
west-2:555555555555:referenceStore/6505293348/reference/5987565360"
     },
     "status": "CREATING",
     "versionName": "my_version"
}
```

Per aggiornare la descrizione di una versione dell'archivio di annotazioni, puoi utilizzare update-annotation-store-versionper aggiungere aggiornamenti a una versione dell'archivio di annotazioni.

```
aws omics update-annotation-store-version \
    --name my_annotation_store \
    --version-name my_version \
    --description "New Description"
```

Riceverai la seguente risposta, che conferma che la versione dell'annotation store è stata aggiornata.

```
{
     "storeId": "4934045d1c6d",
     "id": "2a3f4a44aa7b",
     "description":"New Description",
     "status": "ACTIVE",
     "name": "my_annotation_store",
     "versionName": "my_version",
     "creation Time": "2023-07-21T17:20:59.380043+00:00",
     "updateTime": "2023-07-21T17:26:17.892034+00:00"
}
```

Per visualizzare i dettagli di una versione dell'Annotation Store, usa. get-annotation-store-version

```
aws omics get-annotation-store-version --name my_annotation_store --version-name
 my_version
```

Riceverai una risposta con il nome della versione, lo stato e altri dettagli.

```
{
     "storeId": "4934045d1c6d",
```

```
     "id": "2a3f4a44aa7b",
     "status": "ACTIVE",
     "versionArn": "arn:aws:omics:us-west-2:555555555555:annotationStore/
my_annotation_store/version/my_version",
     "name": "my_annotation_store",
     "versionName": "my_version",
     "creationTime": "2023-07-21T17:15:49.251040+00:00",
     "updateTime": "2023-07-21T17:15:56.434223+00:00",
     "statusMessage": "",
     "versionSizeBytes": 0
    }
```

Per visualizzare tutte le versioni di un archivio di annotazioni, è possibile utilizzare list-annotation-store-versions, come illustrato nell'esempio seguente.

```
aws omics list-annotation-store-versions --name my_annotation_store
```

Riceverai una risposta con le seguenti informazioni

```
{
  "annotationStoreVersions": [
    {
     "storeId": "4934045d1c6d",
     "id": "2a3f4a44aa7b",
     "status": "CREATING",
     "versionArn": "arn:aws:omics:us-west-2:555555555555:annotationStore/
my_annotation_store/version/my_version_2",
     "name": "my_annotation_store",
     "versionName": "my_version_2",
     "creation Time": "2023-07-21T17:20:59.380043+00:00",
     "versionSizeBytes": 0
    },
    {
     "storeId": "4934045d1c6d",
     "id": "4934045d1c6d",
     "status": "ACTIVE",
     "versionArn": "arn:aws:omics:us-west-2:555555555555:annotationStore/
my_annotation_store/version/my_version_1",
     "name": "my_annotation_store",
     "versionName": "my_version_1",
     "creationTime": "2023-07-21T17:15:49.251040+00:00",
     "updateTime": "2023-07-21T17:15:56.434223+00:00",
```

```
     "statusMessage": "",
     "versionSizeBytes": 0
    }
}
```

Se non è più necessaria una versione dell'archivio di annotazioni, è possibile utilizzare delete-annotation-store-versionsper eliminare una versione dell'archivio di annotazioni, come illustrato nell'esempio seguente.

```
aws omics delete-annotation-store-versions --name my_annotation_store --versions
 my_version
```

Se la versione dello store viene eliminata senza errori, riceverai la seguente risposta.

```
{
  "errors": []
}
```

In caso di errori, riceverai una risposta con i dettagli degli errori, come mostrato.

```
{
  "errors": [
    {
      "versionName": "my_version",
      "message": "Version with versionName: my_version was not found."
    }
  ]
}
```

Se tenti di eliminare una versione di Annotation Store con un processo di importazione attivo, riceverai una risposta con un errore, come mostrato.

```
{
  "errors": [
    {
      "versionName": "my_version",
      "message": "version has an inflight import running"
    }
  ]
}
```

In questo caso, è possibile forzare l'eliminazione della versione dell'annotation store, come mostrato nell'esempio seguente.

```
aws omics delete-annotation-store-versions --name my_annotation_store --versions
 my_version --force
```

Eliminazione degli archivi di HealthOmics analisi

> **Important**
>
> AWS HealthOmics i negozi di varianti e i negozi di annotazioni non sono più aperti a nuovi clienti. I clienti esistenti possono continuare a utilizzare il servizio normalmente. Per ulteriori informazioni, consulta AWS HealthOmics modifica della disponibilità dell'archivio delle varianti e dell'archivio delle annotazioni.

Quando si elimina una variante o un archivio di annotazioni, il sistema elimina anche tutti i dati importati in quell'archivio e tutti i tag associati.

L'esempio seguente mostra come eliminare un archivio di varianti utilizzando. AWS CLI Se l'azione ha esito positivo, lo stato dell'archivio delle varianti passa a`DELETING`.

```
aws omics delete-variant-store --id <variant-store-id>
```

L'esempio seguente mostra come eliminare un archivio di annotazioni. Se l'azione ha esito positivo, lo stato dell'archivio di annotazioni passa a. `DELETING` Gli archivi di annotazioni non possono essere eliminati se esiste più di una versione.

```
aws omics delete-annotation-store --id <annotation-store-id>
```

Interrogazione dei HealthOmics dati di analisi

> **Important**
>
> AWS HealthOmics i negozi di varianti e i negozi di annotazioni non sono più aperti a nuovi clienti. I clienti esistenti possono continuare a utilizzare il servizio normalmente. Per ulteriori

informazioni, consulta AWS HealthOmics modifica della disponibilità dell'archivio delle varianti e dell'archivio delle annotazioni.

Puoi eseguire query sui tuoi negozi di varianti utilizzando AWS Lake Formation Amazon Athena o Amazon EMR. Prima di eseguire qualsiasi query, completa le procedure di configurazione (descritte nelle sezioni seguenti) per Lake Formation e Amazon Athena.

Per informazioni su Amazon EMR, consulta il Tutorial: Guida introduttiva ad Amazon EMR

Per gli store di varianti creati dopo il 26 settembre 2024, HealthOmics partiziona il negozio in base all'ID di esempio. Questo partizionamento significa che HealthOmics utilizza l'ID di esempio per ottimizzare la memorizzazione delle informazioni sulle varianti. Le query che utilizzano informazioni di esempio come filtri restituiranno risultati più rapidamente, poiché la query analizza meno dati.

HealthOmics utilizza esempi IDs come nomi di file di partizione. Prima di importare i dati, controllate se l'ID di esempio contiene dati PHI. In caso affermativo, modificate l'ID del campione prima di importare i dati. Per ulteriori informazioni sui contenuti da includere e da non includere nell'esempio IDs, consulta le linee guida sulla pagina web sulla conformità AWS HIPAA.

Argomenti

- Configurazione di Lake Formation per l'uso HealthOmics
- Configurazione di Athena per le interrogazioni
- Esecuzione di interrogazioni sugli store di HealthOmics varianti

Configurazione di Lake Formation per l'uso HealthOmics

Important

AWS HealthOmics i negozi di varianti e i negozi di annotazioni non sono più aperti a nuovi clienti. I clienti esistenti possono continuare a utilizzare il servizio normalmente. Per ulteriori informazioni, consulta AWS HealthOmics modifica della disponibilità dell'archivio delle varianti e dell'archivio delle annotazioni.

Prima di utilizzare Lake Formation per gestire gli archivi HealthOmics dati, esegui le seguenti procedure di configurazione di Lake Formation.

Argomenti

Creazione o verifica degli amministratori di Lake Formation

Prima di poter creare un data lake in Lake Formation, devi definire uno o più amministratori.

Gli amministratori sono utenti e ruoli con le autorizzazioni per creare collegamenti alle risorse. Gli amministratori del data lake vengono configurati per account per regione.

Crea un utente amministratore nella console di Lake Formation

1. Apri la console AWS Lake Formation: console Lake Formation
2. Se la console visualizza il pannello Welcome to Lake Formation, scegli Inizia.

   Lake Formation ti aggiunge alla tabella degli amministratori di Data lake.
3. Altrimenti, dal menu a sinistra, scegli Ruoli e attività amministrative.
4. Aggiungi eventuali amministratori aggiuntivi, se necessario.

Creazione di collegamenti alle risorse utilizzando la console Lake Formation

Per creare una risorsa condivisa su cui gli utenti possano interrogare, i controlli di accesso predefiniti devono essere disabilitati. Per ulteriori informazioni sulla disabilitazione dei controlli di accesso predefiniti, consulta Modifica delle impostazioni di sicurezza predefinite per il tuo data lake nella documentazione di Lake Formation. Puoi creare collegamenti alle risorse individualmente o come gruppo, in modo da poter accedere ai dati in Amazon Athena o in altri AWS servizi (come Amazon EMR).

Creazione di collegamenti alle risorse nella console AWS Lake Formation e condivisione con gli utenti HealthOmics di Analytics

1. Apri la console AWS Lake Formation: console Lake Formation
2. Nella barra di navigazione principale, scegli Database.
3. Nella tabella Database, seleziona il database desiderato.
4. Dal menu Crea, scegli Resource link.

5. Immettete il nome di un link alla risorsa. Se prevedi di accedere al database da Athena, inserisci un nome utilizzando solo lettere minuscole (fino a 256 caratteri).
6. Scegli Create (Crea).
7. Il nuovo collegamento alla risorsa è ora elencato in Database.

Concedi l'accesso alla risorsa condivisa utilizzando la console Lake Formation

Un amministratore del database Lake Formation può concedere l'accesso alla risorsa condivisa utilizzando la seguente procedura.

1. Apri la console AWS Lake Formation: https://console.aws.amazon.com/lakeformation/
2. Nella barra di navigazione principale, scegli Database.
3. Nella pagina Database, seleziona il collegamento alla risorsa creato in precedenza.
4. Dal menu Azioni, scegli Concedi all'obiettivo.
5. Nella pagina Concedi le autorizzazioni per i dati sotto Principali, scegli utenti o ruoli IAM.
6. Dal menu a discesa Utenti o ruoli IAM, trova l'utente a cui desideri concedere l'accesso.
7. Successivamente, nella sezione LF-Tags o nella scheda delle risorse del catalogo, seleziona l'opzione Named data catalog resources.
8. Dal menu a discesa Tables (opzionale), selezionate Tutte le tabelle o la tabella creata in precedenza.
9. Nella scheda Autorizzazioni della tabella, in Autorizzazioni della tabella scegli Descrivi e seleziona.
10. Quindi, scegli Concedi.

Per visualizzare le autorizzazioni di Lake Formation, scegli Autorizzazioni Data lake dal pannello di navigazione principale. La tabella mostra i database e i link alle risorse disponibili.

Configurazione delle autorizzazioni per le condivisioni di risorse AWS RAM

Nella console AWS Lake Formation, visualizza le autorizzazioni scegliendo Autorizzazioni Data lake nella barra di navigazione principale. Nella pagina delle autorizzazioni relative ai dati, puoi visualizzare una tabella che mostra i tipi di risorse, i database e **ARN** che è correlata a una risorsa condivisa in RAM Resource Share. Se devi accettare una condivisione di risorse AWS Resource Access Manager (AWS RAM), ti AWS Lake Formation avvisa nella console.

HealthOmics può accettare implicitamente le condivisioni di AWS RAM risorse durante la creazione del negozio. Per accettare la condivisione di AWS RAM risorse, l'utente o il ruolo IAM che chiama le operazioni `CreateVariantStore` o l'`CreateAnnotationStore`API deve consentire le seguenti azioni:

- `ram:GetResourceShareInvitations`- Questa azione consente di HealthOmics trovare gli inviti.
- `ram:AcceptResourceShareInvitation`- Questa azione consente di HealthOmics accettare l'invito utilizzando un token FAS.

Senza queste autorizzazioni, viene visualizzato un errore di autorizzazione durante la creazione del negozio.

Ecco un esempio di politica che include queste azioni. Aggiungi questa policy all'utente o al ruolo IAM che accetta la condivisione di AWS RAM risorse.

JSON

```
{
  "Version":"2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "omics:*",
        "ram:AcceptResourceShareInvitation",
        "ram:GetResourceShareInvitations"
      ],
      "Resource": "*"
    }
  ]
}
```

Configurazione di Athena per le interrogazioni

> **Important**
>
> AWS HealthOmics i negozi di varianti e i negozi di annotazioni non sono più aperti a nuovi clienti. I clienti esistenti possono continuare a utilizzare il servizio normalmente. Per ulteriori informazioni, consulta AWS HealthOmics modifica della disponibilità dell'archivio delle varianti e dell'archivio delle annotazioni.

Puoi usare Athena per interrogare varianti e annotazioni. Prima di eseguire qualsiasi interrogazione, esegui le seguenti attività di configurazione:

Argomenti

Configurare la posizione dei risultati di una query utilizzando la console Athena

Per configurare la posizione dei risultati di una query, segui questi passaggi.

1. Apri la console Athena: Console Athena
2. Nella barra di navigazione principale, scegli Query editor.
3. Nell'editor di query, scegli la scheda Impostazioni, quindi scegli Gestisci.
4. Inserisci il prefisso S3 di una posizione per salvare il risultato della query.

Configurare un gruppo di lavoro con il motore Athena v3

Per configurare un gruppo di lavoro, segui questi passaggi.

1. Apri la console Athena: Console Athena
2. Nella barra di navigazione principale, scegli Gruppi di lavoro, quindi Crea gruppo di lavoro.
3. Inserisci un nome per il gruppo di lavoro.
4. Seleziona Athena SQL come tipo di motore.
5. In Aggiorna motore di interrogazione, seleziona Manuale.

6. In Query version engine, seleziona Athena versione 3.
7. Selezionare Create workgroup (Crea gruppo di lavoro).

Esecuzione di interrogazioni sugli store di HealthOmics varianti

> **Important**
>
> AWS HealthOmics i negozi di varianti e i negozi di annotazioni non sono più aperti a nuovi clienti. I clienti esistenti possono continuare a utilizzare il servizio normalmente. Per ulteriori informazioni, consulta AWS HealthOmics modifica della disponibilità dell'archivio delle varianti e dell'archivio delle annotazioni.

Puoi eseguire query sul tuo negozio di varianti utilizzando Amazon Athena. Tieni presente che le coordinate genomiche negli archivi di varianti e annotazioni sono rappresentate come intervalli semichiusi, semichiusi e semiaperti.

Esegui una semplice query utilizzando la console Athena

L'esempio seguente mostra come eseguire una query semplice.

1. Aprire l'editor Athena Query: Athena Query editor
2. In Gruppo di lavoro, seleziona il gruppo di lavoro creato durante l'installazione.
3. Verifica che l'origine dati sia. AwsDataCatalog
4. Per Database, seleziona il link alle risorse del database che hai creato durante la configurazione di Lake Formation.
5. Copia la seguente query nel Query Editor nella scheda Query 1:

```
SELECT * from omicsvariants limit 10
```

6. Per eseguire la query, scegli Esegui. La console popola la tabella dei risultati con le prime 10 righe della omicsvariants tabella.

Esegui una query complessa utilizzando la console Athena

L'esempio seguente mostra come eseguire una query complessa. Per eseguire questa query, importala `ClinVar` nell'archivio delle annotazioni.

Esegui un'interrogazione complessa

1. Aprire l'editor Athena Query: Athena Query editor
2. In Gruppo di lavoro, seleziona il gruppo di lavoro creato durante l'installazione.
3. Verifica che l'origine dati sia. AwsDataCatalog
4. Per Database, seleziona il link alle risorse del database che hai creato durante la configurazione di Lake Formation.
5. Scegli l'opzione in alto + a destra per creare una nuova scheda di interrogazione denominata Query 2.
6. Copia la seguente query nel Query Editor nella scheda Query 2:

```
SELECT variants.sampleid,
  variants.contigname,
  variants.start,
  variants."end",
  variants.referenceallele,
  variants.alternatealleles,
  variants.attributes AS variant_attributes,
  clinvar.attributes AS clinvar_attributes
FROM omicsvariants as variants
INNER JOIN omicsannotations as clinvar ON
  variants.contigname=CONCAT('chr',clinvar.contigname)
  AND variants.start=clinvar.start
  AND variants."end"=clinvar."end"
  AND variants.referenceallele=clinvar.referenceallele
  AND variants.alternatealleles=clinvar.alternatealleles
WHERE clinvar.attributes['CLNSIG']='Likely_pathogenic'
```

7. Scegli Esegui per iniziare a eseguire la query.

Condivisione di archivi HealthOmics di analisi

> **Important**
>
> AWS HealthOmics i negozi di varianti e gli archivi di annotazioni non sono più aperti a nuovi clienti. I clienti esistenti possono continuare a utilizzare il servizio normalmente. Per ulteriori informazioni, consulta AWS HealthOmics modifica della disponibilità dell'archivio delle varianti e dell'archivio delle annotazioni.

In qualità di proprietario di un negozio di varianti o di un negozio di annotazioni, puoi condividere lo store con altri account AWS. Il proprietario può revocare l'accesso alla risorsa condivisa eliminando la condivisione.

In qualità di abbonato a un negozio condiviso, devi prima accettare la condivisione. Puoi quindi definire flussi di lavoro che utilizzano l'archivio condiviso. I dati vengono visualizzati sotto forma di tabella sia AWS Glue in Lake Formation che in Lake Formation.

Quando non hai più bisogno di accedere allo store, elimini la condivisione.

Condivisione di risorse tra account in AWS HealthOmicsPer ulteriori informazioni sulla condivisione delle risorse, consulta.

Creazione di una condivisione in negozio

Per creare una condivisione dello store, utilizza l'operazione API create-share. Il sottoscrittore principale è l' Account AWS utente che sottoscriverà la condivisione. L'esempio seguente crea una condivisione per un negozio di varianti. Per condividere un negozio con più di un account, crei più condivisioni dello stesso negozio.

```
aws omics create-share  \
        --resource-arn "arn:aws:omics:us-west-2:555555555555:variantStore/
omics_dev_var_store" \
        --principal-subscriber "123456789012" \
        --name "my_Share-123"
```

Se la creazione ha esito positivo, riceverai una risposta con l'ID e lo stato della condivisione.

```
{
       "shareId": "495c21bedc889d07d0ab69d710a6841e-dd75ab7a1a9c384fa848b5bd8e5a7e0a",
       "name": "my_Share-123",
       "status": "PENDING"
  }
```

La condivisione rimane in sospeso fino a quando il sottoscrittore non la accetta utilizzando l'operazione API accept-share.

Condivisione di risorse tra account in AWS HealthOmics

Utilizza la condivisione tra account per condividere risorse con i collaboratori senza creare copie o modificare le policy delle risorse IAM. Le seguenti risorse supportano la condivisione tra account:

- HealthOmics negozi di varianti
- HealthOmics archivi di annotazioni
- Flussi di lavoro privati

La condivisione di una risorsa include i seguenti passaggi:

1. Il proprietario della risorsa crea una condivisione e specifica l'ARN della risorsa e Account AWS il sottoscrittore previsto. La condivisione di risorse rimane in sospeso fino a quando il sottoscrittore non accetta la condivisione.
2. Il sottoscrittore accetta la condivisione di risorse per accedere alla risorsa. La condivisione delle risorse passa allo stato di attivazione.
3. Il HealthOmics servizio fornisce all'account dell'abbonato l'accesso alla risorsa.
4. Il proprietario della risorsa può eliminare la condivisione oppure il sottoscrittore può revocare l'accesso alla condivisione. Il sottoscrittore non può eliminare la condivisione o la risorsa associata.

Argomenti

Creare una condivisione

Puoi utilizzare l'operazione API create-share per creare una condivisione. Il sottoscrittore principale è l' Account AWS utente che sottoscriverà la risorsa condivisa. L'esempio seguente crea una condivisione per un negozio di varianti.

```
aws omics create-share \
```

```
    --resource-arn "arn:aws:omics:us-west-2:555555555555:variantStore/
omics_dev_var_store" \
    --principal-subscriber "123456789012"  \
    --name "my_Share-123"
```

Se la creazione ha esito positivo, riceverai una risposta con l'ID e lo stato della condivisione.

```
{
"shareId": "495c21bedc889d07d0ab69d710a6841e-dd75ab7a1a9c384fa848b5bd8e5a7e0a",
     "name": "my_Share-123",
     "status": "PENDING"
}
```

La condivisione rimane in sospeso fino a quando il sottoscrittore non la accetta tramite l'operazione accept-share API.

```
 aws omics accept-share \
    --share-id "495c21bedc889d07d0ab69d710a6841e-dd75ab7a1a9c384fa848b5bd8e5a7e0a"
```

Dopo che il sottoscrittore ha accettato la condivisione, la condivisione passa allo stato attivo.

```
{
"status": "ACTIVATING"
}
```

Recupera informazioni su una condivisione

Utilizza l'operazione API get-share per recuperare informazioni sulla condivisione.

```
aws omics get-share --share-id "495c21bedc889d07d0ab69d710a6841e-
dd75ab7a1a9c384fa848b5bd8e5a7e0a"
```

La risposta dell'API include informazioni sui metadati sulla condivisione.

```
{
  "share":
```

```
    {
      "shareId": "495c21bedc889d07d0ab69d710a6841e-dd75ab7a1a9c384fa848b5bd8e5a7e0a",
      "name": "my_Share-123",
      "resourceArn": "arn:aws:omics:us-west-2:555555555555:variantStore/
omics_dev_var_store",
      "principalSubscriber": "123456789012",
      "ownerId": "555555555555",
      "status": "PENDING"
    }
}
```

Visualizza le azioni che possiedi

Utilizza l'API list-shares per recuperare informazioni su ciascuna delle azioni che possiedi.

```
aws omics list-shares  --resource-owner SELF
```

La risposta dell'API include i metadati per ogni condivisione di cui sei proprietario.

Visualizza le condivisioni accettate da altri account

Utilizza l'API list-shares per visualizzare tutte le condivisioni che hai accettato da altri account.

```
aws omics list-shares  --resource-owner OTHER
```

La risposta dell'API include i metadati per ogni condivisione che hai accettato.

Eliminare una condivisione

Utilizza l'API delete-share per eliminare una condivisione dopo che non ti serve più.

```
aws omics delete-share \
    --share-id "495c21bedc889d07d0ab69d710a6841e-dd75ab7a1a9c384fa848b5bd8e5a7e0a"
```

Taggare le risorse in HealthOmics

Argomenti

Avviso importante

HealthOmics protegge i dati dei clienti in base alle policy del modello di responsabilità condivisa di AWS. Ciò significa che tutti i dati dei clienti sono crittografati sia in fase di transizione che in fase di archiviazione. Tuttavia, non tutti i nomi immessi dai clienti per risorse come gli archivi di dati o le operazioni basate sul lavoro sono crittografati. Non devono mai contenere informazioni di identificazione personale o informazioni sanitarie protette. Per ulteriori informazioni, consulta Sicurezza in AWS HealthOmics.

HealthOmics Taggare le risorse

Puoi assegnare metadati alle tue risorse AWS utilizzando i tag. Ogni tag è un'etichetta composta da una chiave e un valore definiti dall'utente. Con i tag è possibile a gestire, identificare, organizzare, cercare e filtrare le risorse.

In questo argomento vengono descritte le categorie e le strategie di tagging comunemente utilizzate per implementare una strategia di tagging coerente ed efficace. Le seguenti sezioni presuppongono una conoscenza di base delle risorse AWS, dei tag, della fatturazione dettagliata e. AWS Identity and Access Management

Ogni tag è costituito da due parti:

- Una chiave di tag (ad esempio CostCenter, Environment o Project). Le chiavi dei tag distinguono tra maiuscole e minuscole

- Un valore di tag (ad esempio, 111122223333 o Production). Come le chiavi dei tag, i valori dei tag distinguono tra maiuscole e minuscole.

È possibile utilizzare i tag per classificare le risorse in base allo scopo, al proprietario, all'ambiente o ad altri criteri. Per ulteriori informazioni, consulta la sezione relativa alle Strategie di tagging di AWS.

È possibile aggiungere, modificare o rimuovere i tag per una risorsa dalla console di servizio della risorsa, dall'API del servizio o dal AWS CLI.

Per abilitare il tagging, assicurati che TagResources sia autorizzato. Puoi autorizzare TagResources allegando una policy IAM come nell'esempio seguente.

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": "omics:Create*",
            "Resource": "*"
        },
        {
            "Effect": "Allow",
            "Action": "omics:Start*",
            "Resource": "*"
        },
        {
            "Effect": "Allow",
            "Action": "omics:Tag*",
            "Resource": "*"
        },
        {
            "Effect": "Allow",
            "Action": "omics:Untag*",
            "Resource": "*"
        },
        {
            "Effect": "Allow",
            "Action": "omics:List*",
            "Resource": "*"
```

```
        }
    ]
}
```

Best practice

Quando crei una strategia di tagging per le risorse AWS, segui le best practice:

- Non archiviare informazioni di identificazione personale (PII), informazioni sanitarie protette (PHI) o altre informazioni sensibili nei tag.
- Utilizza un formato standardizzato con distinzione tra maiuscole e minuscole per i tag e applicalo in modo coerente a tutti i tipi di risorse.
- Prendi in considerazione le linee guida per i tag che supportano più scopi, ad esempio la gestione del controllo dell'accesso alle risorse, il monitoraggio dei costi, l'automazione e l'organizzazione.
- Utilizzate strumenti automatizzati per aiutare a gestire i tag delle risorse. AWS Resource Groups e l'API Resource Groups Tagging consentono il controllo programmatico dei tag, rendendo possibile la gestione, la ricerca e il filtraggio automatici di tag e risorse.
- Il tagging è più efficace quando usi più tag.
- I tag possono essere modificati o modificati in base alle esigenze dell'utente. Tuttavia, per aggiornare i tag di controllo degli accessi, è necessario aggiornare anche le politiche che fanno riferimento a tali tag per controllare l'accesso alle risorse.

Requisiti per il tagging

I tag hanno i requisiti seguenti:

- Le chiavi non possono avere come prefisso aws:.
- Le chiavi devono essere univoche per un set di tag.
- Una chiave deve essere costituita da un numero di caratteri compreso tra 1 e 128.
- Un valore deve essere costituito da un numero di caratteri compreso tra 0 e 256.
- Non è necessario che i valori siano univoci per set di tag.
- I caratteri consentiti per le chiavi e i valori sono lettere Unicode, cifre, spazi e uno qualsiasi dei simboli seguenti: _ . : / = + - @.
- Le chiavi e i valori fanno distinzione tra maiuscole e minuscole.

Sequence Store ha letto i tag del set

Per gli archivi di sequenze, i tag creati sul set di lettura si trovano al livello di risorsa del set di lettura. I set di lettura contengono anche oggetti al loro interno ai quali è possibile accedere, cercare e limitare utilizzando S3 APIs. Per impostazione predefinita, l'ID del campione (OMICS:sampleID) e l'ID del soggetto (OMICS:subjectID) vengono aggiunti all'oggetto.

Inoltre, è possibile sincronizzare fino a cinque tag tra il set di lettura e gli oggetti sottostanti. La configurazione per i tag da sincronizzare è una configurazione a livello di negozio impostata durante la creazione o l'aggiornamento del negozio utilizzando il `propogatedSetLevelTags` parametro.

Se nell'archivio sono già presenti dati, l'aggiornamento delle chiavi potrebbe richiedere del tempo. Durante questo aggiornamento, HealthOmics modifica lo stato del negozio inUpdating. Al termine, HealthOmics imposta lo stato del negozio suActive. Durante la propagazione dei tag, le autorizzazioni basate sui tag potrebbero non essere applicate. Le autorizzazioni verranno applicate dopo il completamento della propagazione dei tag.

Quando i tag vengono impostati o aggiornati sul set di lettura, il sistema decide se aggiornare gli oggetti per quel set di lettura, in base alla configurazione dell'archivio.

Aggiungere un tag a una HealthOmics risorsa

L'aggiunta di tag a una risorsa può aiutarti a identificare e organizzare le tue risorse AWS e a gestirne l'accesso. Innanzitutto, aggiungi uno o più tag (coppie chiave-valore) a una risorsa. Puoi utilizzare fino a 50 tag per risorsa. Esistono anche restrizioni sui caratteri che è possibile utilizzare nei campi chiave e valore.

Dopo aver aggiunto i tag, puoi creare policy IAM per gestire l'accesso alla AWS risorsa in base a questi tag. Puoi usare la HealthOmics console o AWS CLI aggiungere tag a una risorsa. L'aggiunta di tag a un repository può avere impatto sull'accesso a tale repository. Prima di aggiungere un tag a un data store, esamina eventuali policy IAM che potrebbero utilizzare i tag per controllare l'accesso a risorse come gli archivi dati.

I tag di servizio vengono generati automaticamente sia per un oggetto che per un ID di esempio per gli archivi di sequenze.

Segui questi passaggi per utilizzare per aggiungere un tag AWS CLI a una risorsa. HealthOmics Ad esempio, per aggiungere tag a un archivio di sequenze durante la creazione, è possibile utilizzare il

seguente comando in AWS CLI. Il nome dell'archivio di sequenze è MySequenceStore, e i due tag aggiunti con chiavi sono key1 e key2 con valori rispettivamente come value1 e value2 :

```
aws omics create-sequence-store --name "MySequenceStore" --tags key1=value1,key2=value2
```

L'output non elenca i tag. Restituisce la seguente risposta.

```
{
    "id": "6860403586",
    "referenceStoreId": "4889894479",
    "roleArn": "arn:aws:iam::555555555555:role/ImportTest",
    "status": "CREATED",
    "creationTime": "2022-07-21T01:19:07.194Z"
}
```

Per aggiungere tag a una risorsa esistente, è necessario eseguire il seguente comando di esempio.

```
aws omics tag-resource --resource-arn arn:aws:omics:us-
west-2:555555555555:sequenceStore/2275234794 --tags key1=value1,key2=value2
```

In caso di successo, questo comando non restituisce alcuna risposta.

Elenco dei tag per una risorsa

Segui questi passaggi per utilizzare AWS CLI per visualizzare un elenco dei AWS tag di una HealthOmics risorsa. Se non sono stati aggiunti tag, l'elenco restituito è vuoto.

Nel terminale o nella riga di comando, esegui il list-tags-for-resource comando come illustrato nell'esempio seguente.

```
aws omics list-tags-for-resource --resource-arn arn:aws:omics:us-
west-2:555555555555:sequenceStore/2275234794
```

Riceverai un elenco di tag in risposta, in formato JSON.

```
 {
    "tags": {
        "key1": "value1",
```

```
        "key2": "value2"
    }
}
```

Rimozione di tag da un archivio dati

Puoi rimuovere uno o più tag associati a una risorsa. La rimozione di un tag non elimina il tag da altre risorse AWS associate a quel tag.

Nel terminale o nella riga di comando, esegui il comando untag-resource, specificando l'Amazon Resource Name (ARN) della risorsa da cui desideri rimuovere i tag e la chiave del tag che desideri rimuovere.

```
aws omics untag-resource --resource-arn arn:aws:omics:us-
west-2:555555555555:sequenceStore/2275234794 --tag-keys key1,key2
```

In caso di successo, questo comando non restituisce una risposta. Per verificare i tag associati alla risorsa, eseguire il comando list-tags-for-resource.

Autorizzazioni IAM per HealthOmics

Puoi utilizzare AWS Identity and Access Management (IAM) per gestire l'accesso all' HealthOmics API e alle risorse come negozi e flussi di lavoro. Per gli utenti e le applicazioni del tuo account che le utilizzano HealthOmics, gestisci le autorizzazioni in una policy di autorizzazione che puoi applicare a utenti, gruppi o ruoli IAM.

Per gestire le autorizzazioni per gli utenti e le applicazioni nei tuoi account, utilizza le politiche che HealthOmics fornisce o scrivine di tue. La HealthOmics console utilizza più servizi per ottenere informazioni sulla configurazione e sui trigger della funzione. È possibile utilizzare le politiche fornite così come sono o come punto di partenza per politiche più restrittive.

HealthOmics utilizza i ruoli di servizio IAM per accedere ad altri servizi per tuo conto. Ad esempio, puoi creare o scegliere un ruolo di servizio quando esegui un flusso di lavoro che legge i dati da Amazon S3. Per alcune funzionalità, devi anche configurare le autorizzazioni sulle risorse di altri servizi. Esamina questi requisiti prima di iniziare a lavorare con HealthOmics

Per ulteriori informazioni su IAM, consulta Che cos'è IAM? nella Guida per l'utente di IAM.

Argomenti

Politiche IAM basate sull'identità per HealthOmics

Per concedere l'accesso agli utenti del tuo account HealthOmics, utilizzi le politiche basate sull'identità in AWS Identity and Access Management (IAM). Le policy basate sull'identità possono essere applicate direttamente agli utenti IAM o ai gruppi e ruoli IAM associati a un utente. È inoltre possibile concedere le autorizzazioni a utenti in un altro account in modo che assumano un ruolo nel proprio account ed eseguano l'accesso alle risorse HealthOmics.

Per concedere agli utenti l'autorizzazione a eseguire azioni su una versione del flusso di lavoro, è necessario aggiungere il flusso di lavoro e la versione specifica del flusso di lavoro all'elenco delle risorse.

La seguente policy IAM consente a un utente di accedere a tutte le azioni HealthOmics API e di passare i ruoli di servizio a HealthOmics.

Example Policy utente

JSON

```
{
  "Version":"2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "omics:*"
      ],
      "Resource": "*"
    },
    {
      "Effect": "Allow",
      "Action": [
        "iam:PassRole"
      ],
      "Resource": "*",
      "Condition": {
        "StringEquals": {
          "iam:PassedToService": "omics.amazonaws.com"
        }
      }
    }
  ]
}
```

Quando lo utilizzi HealthOmics, interagisci anche con altri AWS servizi. Per accedere a questi servizi, utilizza le politiche gestite fornite da ciascun servizio. Per limitare l'accesso a un sottoinsieme di risorse, puoi utilizzare le policy gestite come punto di partenza per creare politiche personalizzate più restrittive.

- AmazonS3 FullAccess: accesso ai bucket e agli oggetti Amazon S3 utilizzati dai lavori.

- Amazon EC2 ContainerRegistryFullAccess: accesso ai registri e agli archivi Amazon ECR per le immagini dei container del flusso di lavoro.
- AWSLakeFormationDataAdmin— Accesso ai database e alle tabelle di Lake Formation creati dagli archivi di analisi.
- ResourceGroupsandTagEditorFullAccess— Tagga HealthOmics le risorse con operazioni API HealthOmics di tagging.

Le politiche precedenti non consentono a un utente di creare ruoli IAM. Affinché un utente con queste autorizzazioni possa eseguire un job, un amministratore deve creare il ruolo di servizio che concede l' HealthOmics autorizzazione ad accedere alle fonti di dati. Per ulteriori informazioni, consulta Ruoli di servizio per AWS HealthOmics.

Definisci le autorizzazioni IAM personalizzate per le esecuzioni

Puoi includere qualsiasi flusso di lavoro, esecuzione o gruppo di esecuzione a cui fa riferimento la `StartRun` richiesta in una richiesta di autorizzazione. A tale scopo, elenca la combinazione desiderata di flussi di lavoro, esecuzioni o gruppi di esecuzione nella policy IAM. Ad esempio, puoi limitare l'uso di un flusso di lavoro a un'esecuzione o a un gruppo di esecuzioni specifico. È inoltre possibile specificare che un flusso di lavoro venga utilizzato solo con un gruppo di esecuzione.

Di seguito è riportato un esempio di policy IAM che consente un singolo flusso di lavoro con un singolo gruppo di esecuzione.

JSON

```
{
  "Version":"2012-10-17",
  "Statement": [
      {
          "Effect": "Allow",
          "Action": [
              "omics:StartRun"
          ],
          "Resource": [
              "arn:aws:omics:us-west-2:123456789012:workflow/1234567",
              "arn:aws:omics:us-west-2:123456789012:runGroup/2345678"
```

```
            ]
        },
        {
            "Effect": "Allow",
            "Action": [
                "omics:StartRun"
            ],
            "Resource": [
                "arn:aws:omics:us-west-2:123456789012:run/*",
                "arn:aws:omics:us-west-2:123456789012:runGroup/2345678"
            ]
        },
        {
            "Effect": "Allow",
            "Action": [
                "omics:GetRun",
                "omics:ListRunTasks",
                "omics:GetRunTask",
                "omics:CancelRun",
                "omics:DeleteRun"
            ],
            "Resource": [
                "arn:aws:omics:us-west-2:123456789012:run/*"
            ]
        }
    ]
}
```

Ruoli di servizio per AWS HealthOmics

Un ruolo di servizio è un ruolo AWS Identity and Access Management (IAM) che concede le autorizzazioni affinché un AWS servizio acceda alle risorse del tuo account. Assegni un ruolo di servizio a AWS HealthOmics quando avvii un processo di importazione o inizi un'esecuzione.

La HealthOmics console può creare il ruolo richiesto per te. Se utilizzi l' HealthOmics API per gestire le risorse, crea il ruolo di servizio utilizzando la console IAM. Per ulteriori informazioni, consulta Creare un ruolo per delegare le autorizzazioni a un. Servizio AWS

I ruoli di servizio devono avere la seguente politica di attendibilità.

JSON

```
{
  "Version":"2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "Service": "omics.amazonaws.com"
      },
      "Action": "sts:AssumeRole"
    }
  ]
}
```

La politica di fiducia consente al HealthOmics servizio di assumere il ruolo.

Argomenti

- Esempi di politiche di servizio IAM
- CloudFormation Modello di esempio

Esempi di politiche di servizio IAM

In questi esempi, i nomi delle risorse e gli account IDs sono segnaposto da sostituire con valori effettivi.

L'esempio seguente mostra la politica per un ruolo di servizio che è possibile utilizzare per avviare un'esecuzione. La policy concede le autorizzazioni per accedere alla posizione di output di Amazon S3, al gruppo di log del flusso di lavoro e al contenitore Amazon ECR per l'esecuzione.

> **Note**
>
> Se utilizzi la memorizzazione nella cache delle chiamate per l'esecuzione, aggiungi la posizione di run cache Amazon S3 come risorsa nelle autorizzazioni s3.

Example Politica del ruolo di servizio per l'avvio di un'esecuzione

JSON

```
{
"Version":"2012-10-17",
"Statement": [
      {
          "Effect": "Allow",
          "Action": [
              "s3:GetObject",
              "s3:PutObject"
          ],
          "Resource": [
              "arn:aws:s3:::amzn-s3-demo-bucket1/*"
          ]
      },
      {
          "Effect": "Allow",
          "Action": [
              "s3:ListBucket"
          ],
          "Resource": [
              "arn:aws:s3:::amzn-s3-demo-bucket1"
          ]
      },
      {
          "Effect": "Allow",
          "Action": [
              "logs:DescribeLogStreams",
              "logs:CreateLogStream",
              "logs:PutLogEvents"
          ],
          "Resource": [
              "arn:aws:logs:us-east-1:123456789012:log-group:/aws/omics/
WorkflowLog:log-stream:*"
          ]
      },
      {
          "Effect": "Allow",
          "Action": [
              "logs:CreateLogGroup"
          ],
```

```
            "Resource": [
                "arn:aws:logs:us-east-1:123456789012:log-group:/aws/omics/
WorkflowLog:*"
            ]
        },
        {
            "Effect": "Allow",
            "Action": [
                "ecr:BatchGetImage",
                "ecr:GetDownloadUrlForLayer",
                "ecr:BatchCheckLayerAvailability"
            ],
            "Resource": [
                "arn:aws:ecr:us-east-1:123456789012:repository/*"
            ]
        }
    ]
}
```

L'esempio seguente mostra la politica per un ruolo di servizio che è possibile utilizzare per un processo di importazione da un negozio. La policy concede le autorizzazioni per accedere alla posizione di input di Amazon S3.

Example Ruolo di servizio per Reference Store Job

JSON

```
{
"Version":"2012-10-17",
"Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "s3:GetObject"
            ],
            "Resource": [
                "arn:aws:s3:::amzn-s3-demo-bucket/*"
            ]
        },

        {
```

```
            "Effect": "Allow",
            "Action": [
                "s3:GetBucketLocation"
            ],
            "Resource": [
                "arn:aws:s3:::amzn-s3-demo-bucket"
            ]
        }

    ]
}
```

CloudFormation Modello di esempio

Il seguente CloudFormation modello di esempio crea un ruolo di servizio che HealthOmics autorizza ad accedere ai bucket Amazon S3 con nomi preceduti da e a caricare i log del `omics-` flusso di lavoro.

Example Autorizzazioni Reference Store, Amazon S3 e CloudWatch Logs

```
Parameters:
  bucketName:
    Description: Bucket name
    Type: String

Resources:
  serviceRole:
    Type: AWS::IAM::Role
    Properties:
      Policies:
        - PolicyName: read-reference
          PolicyDocument:
            Version: 2012-10-17
            Statement:
            - Effect: Allow
              Action:
                - omics:*
              Resource: !Sub arn:${AWS::Partition}:omics:${AWS::Region}:
${AWS::AccountId}:referenceStore/*
        - PolicyName: read-s3
          PolicyDocument:
            Version: 2012-10-17
```

```
            Statement:
            - Effect: Allow
              Action:
                - s3:ListBucket
              Resource: !Sub arn:${AWS::Partition}:s3:::${bucketName}
            - Effect: Allow
              Action:
                - s3:GetObject
                - s3:PutObject
              Resource: !Sub arn:${AWS::Partition}:s3:::${bucketName}/*
        - PolicyName: upload-logs
          PolicyDocument:
            Version: 2012-10-17
            Statement:
            - Effect: Allow
              Action:
                - logs:DescribeLogStreams
                - logs:CreateLogStream
                - logs:PutLogEvents
              Resource: !Sub arn:${AWS::Partition}:logs:${AWS::Region}:
${AWS::AccountId}:loggroup:/aws/omics/WorkflowLog:log-stream:*
            - Effect: Allow
              Action:
                - logs:CreateLogGroup
              Resource: !Sub arn:${AWS::Partition}:logs:${AWS::Region}:
${AWS::AccountId}:loggroup:/aws/omics/WorkflowLog:*
      AssumeRolePolicyDocument: |
        {
          "Version": "2012-10-17",
          "Statement": [
            {
              "Action": [
                "sts:AssumeRole"
              ],
              "Effect": "Allow",
              "Principal": {
                "Service": [
                  "omics.amazonaws.com"
                ]
              }
            }
          ]
        }
```

Autorizzazioni Amazon ECR

Prima che il HealthOmics servizio possa eseguire un flusso di lavoro in un contenitore dal tuo repository Amazon ECR privato, devi creare una politica delle risorse per il repository. La policy concede l'autorizzazione al HealthOmics servizio di utilizzare il contenitore. Questa politica delle risorse viene aggiunta a ogni repository privato a cui fa riferimento il flusso di lavoro.

> **Note**
>
> L'archivio privato e il flusso di lavoro devono trovarsi nella stessa area.

Se AWS account diversi possiedono il flusso di lavoro e il repository, è necessario configurare le autorizzazioni per più account.

Non è necessario concedere un accesso aggiuntivo al repository per i flussi di lavoro condivisi. Tuttavia, puoi creare politiche che consentano o negano a flussi di lavoro specifici l'accesso all'immagine del contenitore.

Per utilizzare la funzionalità pull through cache di Amazon ECR, devi creare una politica di autorizzazione del registro.

Le seguenti sezioni descrivono come configurare le autorizzazioni delle risorse Amazon ECR per questi scenari. Per ulteriori informazioni sulle autorizzazioni in Amazon ECR, consulta Autorizzazioni di registro privato in Amazon ECR.

Argomenti

Crea una politica delle risorse per il repository Amazon ECR

Crea una politica delle risorse per consentire al HealthOmics servizio di eseguire un flusso di lavoro utilizzando un contenitore nel repository. La policy concede l'autorizzazione al responsabile del HealthOmics servizio di accedere alle azioni Amazon ECR richieste.

Segui questi passaggi per creare la policy:

1. Apri la pagina degli archivi privati nella console Amazon ECR e seleziona il repository a cui concedi l'accesso.
2. Dalla barra di navigazione laterale, seleziona Autorizzazioni.
3. Scegli Modifica.
4. Scegli Modifica policy JSON.
5. Aggiungi la seguente dichiarazione di politica e quindi seleziona Salva.

   JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Sid": "omics workflow access",
            "Effect": "Allow",
            "Principal": {
                "Service": "omics.amazonaws.com"
            },
            "Action": [
                "ecr:GetDownloadUrlForLayer",
                "ecr:BatchGetImage",
                "ecr:BatchCheckLayerAvailability"
            ],
            "Resource": "*"
        }
    ]
}
```

Esecuzione di flussi di lavoro con contenitori multiaccount

Se AWS account diversi possiedono il flusso di lavoro e il contenitore, devi configurare le seguenti autorizzazioni tra account:

1. Aggiorna la policy di Amazon ECR per il repository per concedere esplicitamente l'autorizzazione all'account proprietario del flusso di lavoro.

2. Aggiorna il ruolo di servizio per l'account proprietario del flusso di lavoro per concedergli l'accesso all'immagine del contenitore.

L'esempio seguente dimostra una politica delle risorse di Amazon ECR che concede l'accesso all'account proprietario del flusso di lavoro.

In questo esempio:

- ID account Workflow: 111122223333
- ID dell'account del repository del contenitore: 444455556666
- Nome del contenitore: samtools

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Principal": {
                "Service": "omics.amazonaws.com"
            },
            "Action": [
                "ecr:BatchCheckLayerAvailability",
                "ecr:BatchGetImage",
                "ecr:GetDownloadUrlForLayer"
            ],
            "Resource": "*"
        },
        {
            "Sid": "AllowAccessToTheServiceRoleOfTheAccountThatOwnsTheWorkflow",
            "Effect": "Allow",
            "Principal": {
                "AWS": "arn:aws:iam::111122223333:role/DemoCustomer"
            },
            "Action": [
                "ecr:BatchCheckLayerAvailability",
                "ecr:BatchGetImage",
                "ecr:GetDownloadUrlForLayer"
            ],
```

```
            "Resource": "*"
        }
    ]
}
```

Per completare la configurazione, aggiungi la seguente dichiarazione politica al ruolo di servizio dell'account proprietario del flusso di lavoro. La policy concede il permesso al ruolo di servizio di accedere all'immagine del contenitore «samtools». Assicurati di sostituire i numeri di account, il nome del contenitore e la regione con i tuoi valori.

```
{
    "Sid": "CrossAccountEcrRepoPolicy",
    "Effect": "Allow",
    "Action": ["ecr:BatchCheckLayerAvailability", "ecr:BatchGetImage",
 "ecr:GetDownloadUrlForLayer"],
    "Resource": "arn:aws:ecr:us-west-2:444455556666:repository/samtools"
}
```

Politiche Amazon ECR per flussi di lavoro condivisi

> **Note**
>
> HealthOmics consente automaticamente a un flusso di lavoro condiviso di accedere all'archivio Amazon ECR nell'account del proprietario del flusso di lavoro, mentre il flusso di lavoro è in esecuzione nell'account dell'abbonato. Non è necessario concedere un accesso aggiuntivo al repository per i flussi di lavoro condivisi. Per ulteriori informazioni, consulta Condivisione dei HealthOmics flussi di lavoro.

Per impostazione predefinita, l'abbonato non ha accesso all'archivio Amazon ECR per utilizzare i contenitori sottostanti. Facoltativamente, puoi personalizzare l'accesso al repository Amazon ECR aggiungendo chiavi di condizione alla politica delle risorse del repository. Le seguenti sezioni forniscono esempi di politiche.

Limita l'accesso a flussi di lavoro specifici

È possibile elencare i singoli flussi di lavoro in una dichiarazione condizionale, in modo che solo questi flussi di lavoro possano utilizzare i contenitori nel repository. La chiave di SourceArncondizione

specifica l'ARN del flusso di lavoro condiviso. L'esempio seguente concede l'autorizzazione per il flusso di lavoro specificato per utilizzare questo repository.

JSON

```
{
  "Version":"2012-10-17",
  "Statement": [
     {
       "Sid": "OmicsAccessPrincipal",
       "Effect": "Allow",
       "Principal": {
         "Service": "omics.amazonaws.com"
       },
       "Action": [
         "ecr:GetDownloadUrlForLayer",
         "ecr:BatchGetImage",
         "ecr:BatchCheckLayerAvailability"
       ],
       "Resource": "*",
       "Condition": {
          "StringEquals": {
             "aws:SourceArn": "arn:aws:omics:us-
east-1:111122223333:workflow/1234567"
          }
       }
     }
  ]
}
```

Limita l'accesso a account specifici

È possibile elencare gli account abbonati in una dichiarazione di condizione, in modo che solo tali account abbiano l'autorizzazione a utilizzare i contenitori nel repository. La chiave SourceAccountcondition specifica il sottoscrittore Account AWS . L'esempio seguente concede l'autorizzazione all'account specificato per utilizzare questo repository.

JSON

```
{
```

```
  "Version":"2012-10-17",
  "Statement": [
    {
      "Sid": "OmicsAccessPrincipal",
      "Effect": "Allow",
      "Principal": {
        "Service": "omics.amazonaws.com"
      },
      "Action": [
        "ecr:GetDownloadUrlForLayer",
        "ecr:BatchGetImage",
        "ecr:BatchCheckLayerAvailability"
      ],
      "Resource": "*",
      "Condition": {
        "StringEquals": {
          "aws:SourceAccount": "111122223333"
        }
      }
    }
  ]
}
```

Puoi anche negare le autorizzazioni di Amazon ECR a sottoscrittori specifici, come illustrato nella seguente policy di esempio.

JSON

```
{
      "Version":"2012-10-17",
      "Statement": [
          {
              "Sid": "OmicsAccessPrincipal",
              "Effect": "Allow",
              "Principal": {
                  "Service": "omics.amazonaws.com"
              },
              "Action": [
                  "ecr:GetDownloadUrlForLayer",
                  "ecr:BatchGetImage",
                  "ecr:BatchCheckLayerAvailability"
```

```
            ],
            "Resource": "*",
            "Condition": {
              "StringNotEquals": {
                "aws:SourceAccount": "111122223333"
              }
            }
        }
    ]
}
```

Policy per Amazon ECR pull through cache

Per utilizzare la cache pull through di Amazon ECR, devi creare una politica di autorizzazione del registro. È inoltre possibile creare un modello di creazione di repository, che definisce le autorizzazioni per i repository creati dalla cache pull through di Amazon ECR.

Le sezioni seguenti includono esempi di queste politiche. Per ulteriori informazioni sul pull through cache, consulta Sincronizzare un registro upstream con un registro privato Amazon ECR nella Amazon Elastic Container Registry User Guide.

Politica di autorizzazione del registro

Per utilizzare la cache pull through di Amazon ECR, crea una politica di autorizzazione del registro. La politica di autorizzazione del registro fornisce il controllo sulla replica e consente di recuperare le autorizzazioni della cache.

Per la replica tra più account, è necessario consentire esplicitamente a ciascuno di essi di replicare Account AWS i propri repository nel registro.

Per impostazione predefinita, quando crei una regola pull through cache, qualsiasi principale IAM autorizzato a estrarre immagini da un registro privato può utilizzare anche la regola pull through cache. È possibile utilizzare le autorizzazioni del registro per definire ulteriormente tali autorizzazioni per repository specifici.

Aggiungi una politica di autorizzazione del registro all'account proprietario dell'immagine del contenitore.

Nell'esempio seguente, la policy consente al HealthOmics servizio di creare repository per ogni registro upstream e di avviare richieste pull upstream dai repository creati.

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Sid": "AllowPTCinRegPermissions",
            "Effect": "Allow",
            "Principal": {
                "Service": "omics.amazonaws.com"
            },
            "Action": [
                "ecr:CreateRepository",
                "ecr:BatchImportUpstreamImage"
            ],
            "Resource": [
                "arn:aws:ecr:us-east-1:123456789012:repository/ecr-public/*",
                "arn:aws:ecr:us-east-1:123456789012:repository/docker-hub/*"
            ]
        }
    ]
}
```

Modello di creazione di repository

Per utilizzare il pull through cache in HealthOmics, il repository Amazon ECR deve disporre di un modello di creazione del repository. Il modello definisce le impostazioni di configurazione per i repository privati creati per un registro upstream.

Ogni modello contiene un prefisso dello spazio dei nomi del repository, che Amazon ECR utilizza per abbinare i nuovi repository a un modello specifico. I modelli possono specificare la configurazione per tutte le impostazioni del repository, comprese le policy di accesso basate sulle risorse, l'immutabilità dei tag, la crittografia e le policy del ciclo di vita. Per ulteriori informazioni, consulta Modelli di creazione di repository nella Amazon Elastic Container Registry User Guide.

Nell'esempio seguente, la policy consente al HealthOmics servizio di avviare richieste pull upstream dai repository upstream.

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Sid": "PTCRepoCreationTemplate",
            "Effect": "Allow",
            "Principal": {
                "Service": "omics.amazonaws.com"
            },
            "Action": [
                "ecr:BatchGetImage",
                "ecr:GetDownloadUrlForLayer"
            ],
            "Resource": "*"
        }
    ]
}
```

Politiche per l'accesso ad Amazon ECR su più account

Per l'accesso su più account, il proprietario dell'archivio privato aggiorna la politica di autorizzazione del registro e il modello di creazione del repository per consentire l'accesso all'altro account e al ruolo di esecuzione di quell'account.

Nella politica di autorizzazione del registro, aggiungi un'informativa per consentire al ruolo di esecuzione dell'altro account di accedere alle azioni Amazon ECR:

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Sid": "AllowCrossAccountPTCinRegPermissions",
            "Effect": "Allow",
            "Principal": {
                "AWS": "arn:aws:iam::123456789012:role/RUN_ROLE"},
            "Action": [
```

```
                "ecr:CreateRepository",
                "ecr:BatchGetImage",
                "ecr:BatchImportUpstreamImage"
            ],
            "Resource": "arn:aws:ecr:us-east-1:123456789012:repository/path/*"
        }
    ]
}
```

Nel modello di creazione del repository, aggiungi un'informativa per consentire al ruolo di esecuzione dell'altro account di accedere alle nuove immagini del contenitore. Facoltativamente, puoi aggiungere istruzioni condizionali per limitare l'accesso a flussi di lavoro specifici:

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Sid": "AllowCrossAccountPTCinRepoCreationTemplate",
            "Effect": "Allow",
            "Principal": {
                "AWS": "arn:aws:iam::111122223333:role/RUN_ROLE"},
            "Action": [
                "ecr:BatchGetImage",
                "ecr:GetDownloadUrlForLayer"
            ],
            "Resource": "*",
            "Condition": {
                "StringEquals": {
                    "aws:SourceArn": "arn:aws:omics:us-
east-1:444455556666:workflow/WORKFLOW_ID",
                    "aws:SourceAccount": "111122223333"
                }
            }
        }
    ]
}
```

Aggiungi le autorizzazioni per due azioni aggiuntive (CreateRepository e BatchImportUpstreamImage) nel ruolo di esecuzione e specifica la risorsa a cui il ruolo di esecuzione può accedere.

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Sid": "CrossAccountPTCRunRolePolicy",
            "Effect": "Allow",
            "Action": [
                "ecr:CreateRepository",
                "ecr:BatchImportUpstreamImage",
                "ecr:BatchCheckLayerAvailability",
                "ecr:BatchGetImage",
                "ecr:GetDownloadUrlForLayer",
                "ecr:BatchGetImage"
            ],
            "Resource": "arn:aws:ecr:us-east-1:123456789012::repository/{path}/*"
        }
    ]
}
```

HealthOmics Autorizzazioni per le risorse

AWS HealthOmics crea e accede a risorse in altri servizi per tuo conto quando gestisci un lavoro o crei un negozio. In alcuni casi, è necessario configurare le autorizzazioni in altri servizi per accedere alle risorse o consentire l'accesso HealthOmics ad esse.

Per le autorizzazioni delle risorse relative ad Amazon ECR, consulta. Autorizzazioni Amazon ECR

Autorizzazioni Lake Formation

Prima di utilizzare le funzionalità di analisi in HealthOmics, configura le impostazioni predefinite del database in Lake Formation.

Per configurare le autorizzazioni delle risorse in Lake Formation

1. Apri la pagina delle impostazioni del catalogo dati nella console Lake Formation.
2. Deseleziona i requisiti di controllo degli accessi IAM per database e tabelle in Autorizzazioni predefinite per database e tabelle appena creati.
3. Scegli Save (Salva).

HealthOmics Analytics accetta automaticamente i dati se la policy del servizio dispone delle autorizzazioni RAM corrette, come nell'esempio seguente.

JSON

```
{
  "Version":"2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "omics:*"
      ],
      "Resource": "*"
    },
    {
      "Effect": "Allow",
      "Action": [
        "ram:AcceptResourceShareInvitation",
        "ram:GetResourceShareInvitations"
      ],
      "Resource": "*"
    }
  ]
}
```

Autorizzazioni per l'accesso ai dati tramite Amazon S3 URIs

Puoi accedere ai dati dell'archivio di sequenza utilizzando le operazioni HealthOmics API o le operazioni API di Amazon S3.

Per l'accesso alle HealthOmics API, HealthOmics le autorizzazioni vengono gestite tramite una policy IAM. Tuttavia, S3 Access richiede due livelli di configurazione: autorizzazione esplicita nella policy di accesso S3 dello Store e una policy IAM. Per saperne di più sull'utilizzo delle policy IAM con HealthOmics, consulta Service roles for. HealthOmics

Esistono tre modi per condividere la capacità di leggere oggetti utilizzando Amazon APIs S3:

1. Condivisione basata su policy: questa condivisione richiede l'abilitazione del principio IAM sia nella policy di accesso di S3 che la scrittura di una policy IAM e il suo collegamento al principio IAM. Per maggiori dettagli, consulta l'argomento successivo.
2. Prefirmato URLs : puoi anche generare un URL prefirmato condivisibile per un file nel Sequence Store. Per ulteriori informazioni sulla creazione di prefirmati URLs con Amazon S3, consulta Using URLs presigned nella documentazione di Amazon S3. La policy di accesso di Sequence Store S3 supporta istruzioni per limitare le funzionalità degli URL prefirmati.
3. Ruoli presunti: crea un ruolo all'interno dell'account del proprietario dei dati con una politica di accesso che consenta agli utenti di assumere quel ruolo.

Argomenti

Condivisione basata su policy

Se accedi ai dati dell'archivio di sequenza utilizzando un URI S3 diretto, HealthOmics fornisce misure di sicurezza avanzate per la politica di accesso ai bucket S3 associata.

Le seguenti regole si applicano alle nuove politiche di accesso S3. Per le politiche esistenti, le regole si applicano al successivo aggiornamento della politica:

- Le politiche di accesso di S3 supportano i seguenti elementi di policy
  - Versione, Id, Dichiarazione, Sid, Effetto, Principal, Azione, Risorsa, Condizione
- Le politiche di accesso S3 supportano le seguenti chiavi di condizione:
  - s3:ExistingObjectTag/<key>, s3:prefix, s3:signatureversion, s3: TlsVersion
  - Le policy supportano anche aws: con i seguenti operatori di condizione: e PrincipalArn ArnEquals ArnLike

Se provi ad aggiungere o aggiornare una politica per includere un elemento o una condizione non supportati, il sistema rifiuta la richiesta.

Argomenti

Politica di accesso S3 predefinita

Quando crei un archivio di sequenze, HealthOmics crea una politica di accesso S3 predefinita che concede all'account root del proprietario del data store le seguenti autorizzazioni per tutti gli oggetti accessibili nel sequence store: S3:GetObject, S3 e S3:. GetObjectTagging ListBucket La policy creata di default è:

JSON

```
{
    "Version":"2012-10-17",
    "Statement":
    [
        {
            "Effect": "Allow",
            "Principal":
            {
                "AWS": "arn:aws:iam::111111111111:root"
            },
            "Action":
            [
                "s3:GetObject",
                "s3:GetObjectTagging"
            ],
            "Resource": "arn:aws:s3:us-
west-2:222222222222:accesspoint/111111111111-1234567890/object/111111111111/
sequenceStore/1234567890/*"
        },
        {
            "Effect": "Allow",
```

```
            "Principal":
            {
                "AWS": "arn:aws:iam::111111111111:root"
            },
            "Action": "s3:ListBucket",
            "Resource": "arn:aws:s3:us-
west-2:222222222222:accesspoint/111111111111-1234567890/111111111111/
sequenceStore/1234567890/*"
        }
    ]
}
```

Personalizzazione della politica di accesso

Se la policy di accesso S3 è vuota, non è consentito l'accesso a S3. Se esiste una policy esistente e devi rimuovere l'accesso s3, usa per rimuovere tutti `deleteS3AccessPolicy` gli accessi.

Per aggiungere restrizioni alla condivisione o concedere l'accesso ad altri account, puoi aggiornare la politica utilizzando l'`PutS3AccessPolicy`API. Gli aggiornamenti alla policy non possono andare oltre il prefisso per l'archivio delle sequenze o le azioni specificate.

Policy IAM

Per consentire a un utente o a un principale IAM l'accesso tramite Amazon S3 APIs, oltre all'autorizzazione nella policy di accesso S3, è necessario creare e allegare una policy IAM al principale per concedere l'accesso. Una policy che consente l'accesso all'API Amazon S3 può essere applicata a livello di archivio di sequenza o a livello di set di lettura. A livello di set di lettura, l'autorizzazione può essere limitata tramite il prefisso o utilizzando filtri di tag di risorsa per modelli di ID campione o soggetto.

Se l'archivio delle sequenze utilizza una chiave gestita dal cliente (CMK), il principale deve disporre anche dei diritti per utilizzare la chiave KMS per la decrittografia. Per ulteriori informazioni, consulta Accesso KMS su più account nella Guida per gli sviluppatori. AWS Key Management Service

L'esempio seguente fornisce a un utente l'accesso a un archivio di sequenze. È possibile ottimizzare l'accesso con condizioni aggiuntive o filtri basati sulle risorse.

JSON

```
{
```

```
  "Version":"2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "AWS": "arn:aws:iam::111111111111:root"
      },
      "Action":
      [
         "s3:GetObject",
         "s3:GetObjectTagging"
      ],
      "Resource": "arn:aws:s3:us-
west-2:222222222222:accesspoint/111111111111-1234567890/object/111111111111/
sequenceStore/1234567890/*",
      "Condition": {
        "StringEquals": {
          "s3:ExistingObjectTag/omics:readSetStatus": "ACTIVE"
        }
      }
    },
    {
      "Effect": "Allow",
      "Principal": {
        "AWS": "arn:aws:iam::111111111111:root"
      },
      "Action": "s3:ListBucket",
      "Resource": "arn:aws:s3:us-
west-2:222222222222:accesspoint/111111111111-1234567890",
      "Condition": {
        "StringLike": {
          "s3:prefix": "111111111111/sequenceStore/1234567890/*"
        }
      }
    }
  ]
}
```

Controllo degli accessi basato su tag

Per utilizzare il controllo degli accessi basato su tag, è necessario prima aggiornare l'archivio delle sequenze per propagare le chiavi dei tag che verranno utilizzate. Questa configurazione viene

impostata durante la creazione o l'aggiornamento dell'archivio di sequenze. Una volta che i tag sono stati propagati, è possibile utilizzare le condizioni dei tag per aggiungere ulteriori restrizioni. Le restrizioni possono essere inserite nella policy di S3 Access o nella policy IAM. Di seguito è riportato un esempio di policy di accesso S3 basata su schede che verrebbe impostata:

```
{
    "Sid": "tagRestrictedGets",
    "Effect": "Allow",
    "Principal":
    {
        "AWS": "arn:aws:iam::<target_restricted_account_id>:root"
    },
    "Action":
    [
        "s3:GetObject",
        "s3:GetObjectTagging"
    ],
    "Resource": "arn:aws:s3:us-west-2:222222222222:accesspoint/111111111111-1234567890/
object/111111111111/sequenceStore/1234567890/*",
    "Condition":
    {
        "StringEquals":
        {
            "s3:ExistingObjectTag/tagKey1": "tagValue1",
            "s3:ExistingObjectTag/tagKey2": "tagValue2"
        }
    }
}
```

Esempio di restrizione

Scenario: creazione di una condivisione in cui il proprietario dei dati può limitare la capacità di un utente di scaricare i dati «ritirati».

In questo scenario, un proprietario dei dati (account #111111111111) gestiva un archivio dati. Questo proprietario dei dati condivide i dati con un'ampia gamma di utenti terzi, tra cui un ricercatore (account #999999999999). Nell'ambito della gestione dei dati, il proprietario dei dati riceve periodicamente richieste di ritiro dei dati di un partecipante. Per gestire questa revoca, il proprietario dei dati limita innanzitutto l'accesso diretto al download dopo aver ricevuto la richiesta e infine elimina i dati in base alle proprie esigenze.

Per soddisfare questa esigenza, il proprietario dei dati configura un archivio di sequenze e ogni set di lettura riceve un tag per lo «stato» che verrà impostato su «ritirato» se la richiesta di prelievo arriva. Per i dati con il tag impostato su questo valore, vogliono assicurarsi che nessun utente possa eseguire «getObject» su questo file. Per eseguire questa configurazione, il proprietario dei dati dovrà assicurarsi che vengano eseguiti due passaggi.

Passaggio 1. Per l'archivio delle sequenze, assicuratevi che il tag di stato sia aggiornato per essere propagato. Questo viene fatto aggiungendo la chiave «status» nel campo «`propogatedSetLevelTags`when call» o `createSequenceStore` `updateSequenceStore.`

Passaggio 2. Aggiorna la politica di accesso s3 dello store per limitare getObject agli oggetti con il tag di stato impostato su withdrawned. Questo viene fatto aggiornando la politica di accesso ai negozi utilizzando l'API. `PutS3AccesPolicy` La seguente politica consentirebbe ai clienti di continuare a visualizzare i file ritirati quando elencano gli oggetti, ma impedirebbe loro di accedervi:

- Dichiarazione 1 (restrictedGetWithdrawal): L'account 9999 non può recuperare gli oggetti che vengono ritirati.
- Dichiarazione 2 (ownerGetAll): L'account 1111, il proprietario dei dati, può recuperare tutti gli oggetti, inclusi gli oggetti che vengono ritirati.
- Dichiarazione 3 (everyoneListAll): Tutti gli account condivisi, +1 1111 e «9999», possono eseguire l'ListBucketoperazione sull'intero prefisso.

JSON

```
{
    "Version":"2012-10-17",
    "Statement":
    [
        {
            "Sid": "restrictedGetWithdrawal",
            "Effect": "Allow",
            "Principal":
            {
                "AWS": "arn:aws:iam::999999999999:root"
            },
            "Action":
            [
                "s3:GetObject",
                "s3:GetObjectTagging"
```

```
            ],
            "Resource": "arn:aws:s3:us-
west-2:222222222222:accesspoint/111111111111-1234567890/object/111111111111/
sequenceStore/1234567890/*",
            "Condition":
            {
                "StringNotEquals":
                {
                    "s3:ExistingObjectTag/status": "withdrawn"
                }
            }
        },
        {
            "Sid": "ownerGetAll",
            "Effect": "Allow",
            "Principal":
            {
                "AWS": "arn:aws:iam::111111111111:root"
            },
            "Action":
            [
                "s3:GetObject",
                "s3:GetObjectTagging"
            ],
            "Resource": "arn:aws:s3:us-
west-2:222222222222:accesspoint/111111111111-1234567890/object/111111111111/
sequenceStore/1234567890/*",
            "Condition":
            {
                "StringEquals":
                {
                    "s3:ExistingObjectTag/omics:readSetStatus": "ACTIVE"
                }
            }
        },
        {
            "Sid": "everyoneListAll",
            "Effect": "Allow",
            "Principal":
            {
                "AWS": [
                    "arn:aws:iam::111111111111:root",
                    "arn:aws:iam::999999999999:root"
                ]
```

```
            },
            "Action": "s3:ListBucket",
            "Resource": "arn:aws:s3:us-
west-2:222222222222:accesspoint/111111111111-1234567890",
            "Condition":
            {
                "StringLike":
                {
                    "s3:prefix": "111111111111/sequenceStore/1234567890/*"
                }
            }
        }
    ]
}
```

Sicurezza in AWS HealthOmics

La sicurezza del cloud AWS è la massima priorità. In qualità di AWS cliente, puoi beneficiare di data center e architetture di rete progettati per soddisfare i requisiti delle organizzazioni più sensibili alla sicurezza.

La sicurezza è una responsabilità condivisa tra te e te. AWS Il modello di responsabilità condivisa descrive questo aspetto come sicurezza del cloud e sicurezza nel cloud:

- Sicurezza del cloud: AWS è responsabile della protezione dell'infrastruttura che gestisce AWS i servizi in Cloud AWS. AWS fornisce inoltre servizi che è possibile utilizzare in modo sicuro. I revisori esterni testano e verificano regolarmente l'efficacia della nostra sicurezza nell'ambito dei AWS Programmi di AWS conformità dei Programmi di conformità dei di . Per ulteriori informazioni sui programmi di conformità applicabili ad AWS HealthOmics, consulta AWS Services in Scope by Compliance Program AWS .
- Sicurezza nel cloud: la tua responsabilità è determinata dal AWS servizio che utilizzi. L'utente è anche responsabile di altri fattori, tra cui la riservatezza dei dati, i requisiti della propria azienda e le leggi e normative vigenti.

Questa documentazione ti aiuta a capire come applicare il modello di responsabilità condivisa quando usi AWS HealthOmics. I seguenti argomenti mostrano come configurare AWS per HealthOmics soddisfare i tuoi obiettivi di sicurezza e conformità. Imparerai anche a usare altri AWS servizi che ti aiutano a monitorare e proteggere le tue HealthOmics risorse AWS.

Argomenti

Protezione dei dati in AWS HealthOmics

Il modello di responsabilità AWS condivisa modello di si applica alla protezione dei dati in AWS HealthOmics. Come descritto in questo modello, AWS è responsabile della protezione

dell'infrastruttura globale che gestisce tutto il Cloud AWS. L'utente è responsabile del controllo dei contenuti ospitati su questa infrastruttura. L'utente è inoltre responsabile della configurazione della protezione e delle attività di gestione per i Servizi AWS utilizzati. Per maggiori informazioni sulla privacy dei dati, consulta le Domande frequenti sulla privacy dei dati. Per informazioni sulla protezione dei dati in Europa, consulta il post del blog relativo al AWS Modello di responsabilità condivisa e GDPR nel AWS Blog sulla sicurezza.

Ai fini della protezione dei dati, consigliamo di proteggere Account AWS le credenziali e configurare i singoli utenti con AWS IAM Identity Center or AWS Identity and Access Management (IAM). In tal modo, a ogni utente verranno assegnate solo le autorizzazioni necessarie per svolgere i suoi compiti. Suggeriamo, inoltre, di proteggere i dati nei seguenti modi:

- Utilizza l'autenticazione a più fattori (MFA) con ogni account.
- SSL/TLS Da utilizzare per comunicare con AWS le risorse. È richiesto TLS 1.2 ed è consigliato TLS 1.3.
- Configura l'API e la registrazione delle attività degli utenti con AWS CloudTrail. Per informazioni sull'utilizzo dei CloudTrail percorsi per acquisire AWS le attività, consulta Lavorare con i CloudTrail percorsi nella Guida per l'AWS CloudTrail utente.
- Utilizza soluzioni di AWS crittografia, insieme a tutti i controlli di sicurezza predefiniti all'interno Servizi AWS.
- Utilizza i servizi di sicurezza gestiti avanzati, come Amazon Macie, che aiutano a individuare e proteggere i dati sensibili archiviati in Amazon S3.
- Se hai bisogno di moduli crittografici convalidati FIPS 140-3 per accedere AWS tramite un'interfaccia a riga di comando o un'API, usa un endpoint FIPS. Per ulteriori informazioni sugli endpoint FIPS disponibili, consulta il Federal Information Processing Standard (FIPS) 140-3.

Ti consigliamo di non inserire mai informazioni riservate o sensibili, ad esempio gli indirizzi e-mail dei clienti, nei tag o nei campi di testo in formato libero, ad esempio nel campo Nome. Ciò include quando lavori con AWS HealthOmics o altri Servizi AWS utenti utilizzando la console, l'API o AWS SDKs. AWS CLI I dati inseriti nei tag o nei campi di testo in formato libero utilizzati per i nomi possono essere utilizzati per i la fatturazione o i log di diagnostica. Quando si fornisce un URL a un server esterno, suggeriamo vivamente di non includere informazioni sulle credenziali nell'URL per convalidare la richiesta al server.

Crittografia dei dati a riposo

Argomenti

Per proteggere i dati sensibili dei clienti archiviati, AWS HealthOmics fornisce la crittografia di default utilizzando una chiave AWS Key Management Service (AWS KMS) di proprietà del servizio. Sono supportate anche le chiavi gestite dal cliente. Per ulteriori informazioni sulla chiave gestita dal cliente, consulta Amazon Key Management Service.

Tutti gli archivi HealthOmics dati (Storage e Analytics) supportano l'uso di chiavi gestite dal cliente. La configurazione di crittografia non può essere modificata dopo la creazione di un archivio dati. Se un data store utilizza un Chiave di proprietà di AWS, verrà contrassegnato come «inattivo» AWS_OWNED_KMS_KEY e la chiave specifica utilizzata per la crittografia non sarà visibile.

Per i HealthOmics flussi di lavoro, le chiavi gestite dal cliente non sono supportate dal file system temporaneo; tuttavia, tutti i dati inattivi vengono crittografati automaticamente utilizzando l'algoritmo di crittografia a blocchi XTS-AES-256 per crittografare il file system. L'utente e il ruolo IAM utilizzati per avviare l'esecuzione di un flusso di lavoro devono inoltre avere accesso alle chiavi utilizzate per i bucket di input e output del flusso di lavoro. AWS KMS I flussi di lavoro non utilizzano sovvenzioni e la AWS KMS crittografia è limitata ai bucket Amazon S3 di input e output. Il ruolo IAM utilizzato sia per il flusso di lavoro APIs deve avere accesso alle AWS KMS chiavi utilizzate sia ai bucket di input e output di Amazon S3. Puoi utilizzare i ruoli e le autorizzazioni IAM per controllare l'accesso o le politiche. AWS KMS Per saperne di più, consulta Autenticazione e controllo degli accessi per AWS KMS.

Quando utilizzi AWS Lake Formation HealthOmics Analytics, tutte le autorizzazioni di decrittografia associate a Lake Formation vengono assegnate anche ai bucket Amazon S3 di input e output. Ulteriori informazioni su come AWS Lake Formation gestire le autorizzazioni sono disponibili nella documentazione.AWS Lake Formation

HealthOmics Analytics concede a Lake Formation kms: Decrypt le autorizzazioni per leggere i dati crittografati in un bucket Amazon S3. Finché disponi delle autorizzazioni per interrogare i dati tramite

Lake Formation, sarai in grado di leggere i dati crittografati. L'accesso ai dati è controllato tramite il controllo dell'accesso ai dati in Lake Formation, non tramite una politica chiave KMS. Per ulteriori informazioni, consulta le richieste di servizi AWS AWS integrate nella documentazione di Lake Formation.

Chiavi di proprietà di AWS

Per impostazione predefinita, HealthOmics Chiavi di proprietà di AWS crittografa automaticamente i dati inattivi, poiché questi dati possono contenere informazioni sensibili come informazioni di identificazione personale (PII) o Protected Health Information (PHI). Chiavi di proprietà di AWS non sono archiviati nel tuo account. Fanno parte di una raccolta di chiavi KMS di proprietà e gestione di AWS da utilizzare in più account AWS.

I servizi AWS possono essere utilizzati Chiavi di proprietà di AWS per proteggere i tuoi dati. Non puoi visualizzarne, gestirli Chiavi di proprietà di AWS, accedervi o verificarne l'utilizzo. Tuttavia, non è necessario eseguire alcuna operazione o modificare alcun programma per proteggere le chiavi che crittografano i dati.

Non ti viene addebitata una tariffa mensile o una tariffa di utilizzo per l'utilizzo Chiavi di proprietà di AWS e non vengono conteggiate nelle quote AWS KMS per il tuo account. Per ulteriori informazioni, consulta Chiavi gestite da AWS.

Chiavi gestite dal cliente

HealthOmics supporta l'uso di chiavi simmetriche gestite dal cliente che crei, possiedi e gestisci per aggiungere un secondo livello di crittografia rispetto alla crittografia esistente di proprietà di AWS. Avendo il pieno controllo di questo livello di crittografia, è possibile eseguire operazioni quali:

- Stabilire e mantenere politiche chiave, politiche IAM e sovvenzioni
- Ruotare i materiali crittografici delle chiavi
- Abilitare e disabilitare le policy delle chiavi
- Aggiungere tag
- Creare alias delle chiavi
- Pianificare l'eliminazione delle chiavi

Puoi anche utilizzarlo CloudTrail per tenere traccia delle richieste HealthOmics inviate a per tuo AWS KMS conto. AWS KMS Si applicano costi aggiuntivi. Per ulteriori informazioni, consulta Customer Managed Keys.

Creazione di una chiave gestita dal cliente

Puoi creare una chiave simmetrica gestita dal cliente utilizzando la Console di gestione AWS o il. AWS KMS APIs

Segui i passaggi per la creazione di chiavi simmetriche gestite dal cliente nella AWS Key Management Service Developer Guide.

Le policy della chiave controllano l'accesso alla chiave gestita dal cliente. Ogni chiave gestita dal cliente deve avere esattamente una policy della chiave, che contiene istruzioni che determinano chi può usare la chiave e come la possono usare. Quando crei una chiave gestita dal cliente, puoi specificare una policy chiave. Per ulteriori informazioni, consulta Managing access to customer managed keys nella AWS Key Management Service Developer Guide.

Per utilizzare una chiave gestita dal cliente con le tue risorse HealthOmics Analytics, il mittente chiamante richiede kms: CreateGrant operations nella policy chiave. Ciò consente al sistema di utilizzare un token FAS per creare una concessione a una chiave gestita dal cliente che controlla l'accesso a una chiave KMS specificata. Questa chiave consente a un utente di accedere alle operazioni kms:grant richieste. HealthOmics Per ulteriori informazioni, vedere Utilizzo delle sovvenzioni.

Per l' HealthOmics analisi, devono essere consentite le seguenti operazioni API per il principale chiamante:

- kms: CreateGrant aggiunge le concessioni a una chiave gestita dal cliente specifica, che consente l'accesso alle operazioni di concessione in HealthOmics Analytics.
- kms: DescribeKey fornisce i dettagli chiave gestiti dal cliente necessari per convalidare la chiave. Questo è necessario per tutte le operazioni.
- kms: GenerateDataKey fornisce l'accesso alle risorse di crittografia a riposo per tutte le operazioni di scrittura. Inoltre, questa azione fornisce dettagli chiave gestiti dal cliente che il servizio può utilizzare per verificare che il chiamante abbia accesso all'utilizzo della chiave.
- KMS:Decrypt fornisce l'accesso alle operazioni di lettura o ricerca per risorse crittografate.

Per utilizzare una chiave gestita dal cliente con le risorse HealthOmics di archiviazione, il responsabile del HealthOmics servizio e il principale chiamante devono essere consentiti nella politica chiave. Ciò consente al servizio di verificare che il chiamante abbia accesso alla chiave e utilizza il responsabile del servizio per eseguire la gestione del negozio utilizzando la chiave gestita

dal cliente. Per quanto riguarda l' HealthOmics archiviazione, la politica chiave per il responsabile del servizio deve consentire le seguenti operazioni API:

- kms: DescribeKey fornisce i dettagli chiave gestiti dal cliente necessari per convalidare la chiave. Questo è necessario per tutte le operazioni.
- kms: GenerateDataKey fornisce l'accesso alle risorse di crittografia a riposo per tutte le operazioni di scrittura. Inoltre, questa azione fornisce dettagli chiave gestiti dal cliente che il servizio può utilizzare per verificare che il chiamante abbia accesso all'utilizzo della chiave.
- KMS:Decrypt fornisce l'accesso alle operazioni di lettura o ricerca per risorse crittografate.

L'esempio seguente mostra una dichiarazione politica che consente a un responsabile del servizio di creare e interagire con una HealthOmics sequenza o un archivio di riferimento crittografato utilizzando la chiave gestita dal cliente:

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
{
    "Effect": "Allow",
    "Principal": {
        "Service": "omics.amazonaws.com"
    },
    "Action": [
        "kms:Decrypt",
        "kms:DescribeKey",
        "kms:Encrypt",
        "kms:GenerateDataKey*"
    ],
    "Resource": "*"
        }
    ]
}
```

L'esempio seguente mostra una policy che crea autorizzazioni per un data store per decrittografare i dati da un bucket Amazon S3.

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "omics:GetReference",
                "omics:GetReferenceMetadata"
            ],
            "Resource": [
                "arn:aws:omics:us-east-1:123456789012:referenceStore/*"
            ]
        },
        {
            "Effect": "Allow",
            "Action": [
                "s3:GetObject"
            ],
            "Resource": [
                "arn:aws:s3:::[[s3path]]/*"
            ]
        },
        {
            "Effect": "Allow",
            "Action": [
                "kms:Decrypt"
            ],
            "Resource": [
                "arn:aws:kms:us-east-1:123456789012:key/key_id"
            ],
            "Condition": {
              "StringEquals": {
                  "kms:ViaService": [
                    "s3.us-east-1.amazonaws.com"
                  ]
              }
           }
        }
    ]
}
```

Autorizzazioni IAM richieste per l'utilizzo di una chiave gestita dal cliente

Quando si crea una risorsa come un archivio dati con AWS KMS crittografia utilizzando una chiave gestita dal cliente, sono necessarie le autorizzazioni sia per la policy chiave che per la policy IAM per l'utente o il ruolo IAM.

Puoi utilizzare la chiave kms: ViaService condition per limitare l'uso della chiave KMS solo alle richieste che provengono da. HealthOmics

Per ulteriori informazioni sulle policy chiave, consulta Enabling IAM policies nella AWS Key Management Service Developer Guide.

Argomenti

Autorizzazioni dell'API di analisi

Per l' HealthOmics analisi, l'utente o il ruolo IAM che crea gli store deve disporre delle autorizzazioni kms:CreateGrant, kms:GenerateDataKey, kms: Decrypt e e kms: DescribeKey delle autorizzazioni necessarie. HealthOmics

Autorizzazioni dell'API di archiviazione

Per HealthOmics lo storage APIs, l'utente o il ruolo IAM che richiama le seguenti operazioni API richiede le autorizzazioni elencate:

CreateReferenceStore, CreateSequenceStore

Per creare un negozio, il chiamante IAM deve disporre `kms:DescribeKey` delle autorizzazioni più le autorizzazioni necessarie. HealthOmics Il principale del HealthOmics servizio chiama `kms:GenerateDataKeyWithoutPlaintext` per eseguire controlli di convalida dell'accesso per il caricamento e l'accesso ai dati.

StartReadSetImportJob, StartReferenceImportJob

Per avviare i processi di importazione dei dati, il chiamante IAM deve disporre `kms:Decrypt` `kms:GenerateDataKey` delle autorizzazioni per la chiave KMS nello store per l'importazione e

delle `kms:Decrypt` autorizzazioni per il bucket Amazon S3 contenente gli oggetti da importare. Inoltre, il ruolo passato alla chiamata deve disporre delle `kms:Decrypt` autorizzazioni sul bucket Amazon S3 contenente gli oggetti da importare. Il chiamante IAM deve inoltre disporre delle autorizzazioni per passare il ruolo al job.

CreateMultipartReadSetUpload, UploadReadSetPart, CompleteMultipartReadSetUpload

Per completare un caricamento in più parti, il chiamante IAM deve avere `kms:Decrypt` e deve creare, `kms:GenerateDataKey` caricare e completare il caricamento in più parti.

StartReadSetExportJob

Per avviare un processo di esportazione dei dati, il chiamante IAM deve disporre dell'`kms:Decrypt`autorizzazione per l'esportazione della chiave KMS sullo store da `kms:GenerateDataKey` e `kms:Decrypt` delle autorizzazioni sul bucket Amazon S3 che riceve gli oggetti. Inoltre, il ruolo passato alla chiamata deve disporre delle `kms:Decrypt` autorizzazioni sul bucket Amazon S3 che riceve gli oggetti. Il chiamante IAM deve inoltre disporre delle autorizzazioni per passare il ruolo al job.

StartReadsetActivationJob

Per avviare un processo di attivazione del set di lettura, il chiamante IAM deve disporre delle autorizzazioni `kms:Decrypt` e dei `kms:GenerateDataKey` permessi per gli oggetti.

GetReference, GetReadSet

Per leggere gli oggetti dallo store, il chiamante IAM deve disporre dell'`kms:Decrypt`autorizzazione per gli oggetti.

Leggi Set S3 GetObject

Per leggere gli oggetti dallo store utilizzando l'`GetObject`API Amazon S3, il chiamante IAM deve disporre dell'`kms:Decrypt`autorizzazione per gli oggetti. Imposta questa autorizzazione sia per la chiave gestita dal cliente che per le configurazioni Chiave di proprietà di AWS .

Come HealthOmics utilizza le sovvenzioni in AWS KMS

HealthOmics Analytics richiede una concessione per utilizzare la chiave KMS gestita dal cliente. Le sovvenzioni non sono richieste o utilizzate per HealthOmics i flussi di lavoro. HealthOmics Lo storage utilizza la chiave gestita dal cliente direttamente dal responsabile del servizio, quindi non utilizzare una concessione. Quando crei un archivio di analisi crittografato con una chiave gestita dal cliente, HealthOmics Analytics crea una concessione per tuo conto inviando una CreateGrantrichiesta ad

AWS KMS. Le sovvenzioni in AWS KMS vengono utilizzate per consentire l' HealthOmics accesso a una chiave KMS in un account cliente.

Non è consigliabile revocare o ritirare le sovvenzioni che Analytics crea per tuo conto. HealthOmics Se revochi o ritiri la concessione che HealthOmics autorizza l'uso delle chiavi AWS KMS nel tuo account, HealthOmics non puoi accedere a questi dati, crittografare nuove risorse trasferite nel data store o decrittografarle quando vengono estratte.

Quando revochi o ritiri una sovvenzione per, la modifica avviene immediatamente. HealthOmics Per revocare i diritti di accesso, ti consigliamo di eliminare l'archivio dati anziché revocare la concessione. Quando elimini il data store, annulla le HealthOmics concessioni per tuo conto.

Monitoraggio delle chiavi di crittografia per AWS HealthOmics

Puoi utilizzarlo CloudTrail per tenere traccia delle richieste AWS HealthOmics inviate a per tuo AWS KMS conto quando utilizzi una chiave gestita dal cliente. Le voci di registro nel CloudTrail registro mostrano HealthOmics .amazonaws.com nel campo UserAgent per distinguere chiaramente le richieste effettuate da. HealthOmics

Gli esempi seguenti sono CloudTrail eventi per CreateGrant GenerateDataKey, Decrypt e per monitorare AWS KMS le operazioni richiamate per accedere DescribeKey HealthOmics ai dati crittografati dalla chiave gestita dal cliente.

Di seguito viene inoltre illustrato come utilizzare CreateGrant per consentire l' HealthOmics analisi di accedere a una chiave KMS fornita dal cliente, in modo HealthOmics da utilizzare tale chiave KMS per crittografare tutti i dati inattivi del cliente.

Non è necessario creare le proprie sovvenzioni. HealthOmics crea una sovvenzione per tuo conto inviando una CreateGrant richiesta ad AWS KMS. Le sovvenzioni AWS KMS vengono utilizzate per HealthOmics consentire l'accesso a una AWS KMS chiave in un account cliente.

```
{
    "eventVersion": "1.08",
    "userIdentity": {
        "type": "AssumedRole",
        "principalId": "xx:test",
        "arn": "arn:AWS:sts::555555555555:assumed-role/user-admin/test",
        "accountId": "xx",
        "accessKeyId": "xxx",
        "sessionContext": {
            "sessionIssuer": {
                "type": "Role",
```

```
                "principalId": "xxxx",
                "arn": "arn:AWS:iam::555555555555:role/user-admin",
                "accountId": "555555555555",
                "userName": "user-admin"
            },
            "webIdFederationData": {},
            "attributes": {
                "creationDate": "2022-11-11T01:36:17Z",
                "mfaAuthenticated": "false"
            }
        },
        "invokedBy": "apigateway.amazonAWS.com"
    },
    "eventTime": "2022-11-11T02:34:41Z",
    "eventSource": "kms.amazonAWS.com",
    "eventName": "CreateGrant",
    "AWSRegion": "us-west-2",
    "sourceIPAddress": "apigateway.amazonAWS.com",
    "userAgent": "apigateway.amazonAWS.com",
    "requestParameters": {
        "granteePrincipal": "AWS Internal",
        "keyId": "arn:AWS:kms:us-west-2:555555555555:key/a6e87d77-cc3e-4a98-a354-
e4c275d775ef",
        "operations": [
            "CreateGrant",
            "RetireGrant",
            "Decrypt",
            "GenerateDataKey"
        ]
    },
    "responseElements": {
        "grantId": "4869b81e0e1db234342842af9f5531d692a76edaff03e94f4645d493f4620ed7",
        "keyId": "arn:AWS:kms:us-west-2:245126421963:key/xx-cc3e-4a98-a354-
e4c275d775ef"
    },
    "requestID": "d31d23d6-b6ce-41b3-bbca-6e0757f7c59a",
    "eventID": "3a746636-20ef-426b-861f-e77efc56e23c",
    "readOnly": false,
    "resources": [
        {
            "accountId": "245126421963",
            "type": "AWS::KMS::Key",
            "ARN": "arn:AWS:kms:us-west-2:245126421963:key/xx-cc3e-4a98-a354-
e4c275d775ef"
```

```
        }
    ],
    "eventType": "AWSApiCall",
    "managementEvent": true,
    "recipientAccountId": "245126421963",
    "eventCategory": "Management"
}
```

L'esempio seguente mostra come GenerateDataKey garantire che l'utente disponga delle autorizzazioni necessarie per crittografare i dati prima di archiviarli.

```
{
    "eventVersion": "1.08",
    "userIdentity": {
        "type": "AssumedRole",
        "principalId": "EXAMPLEUSER",
        "arn": "arn:AWS:sts::111122223333:assumed-role/Sampleuser01",
        "accountId": "111122223333",
        "accessKeyId": "EXAMPLEKEYID",
        "sessionContext": {
            "sessionIssuer": {
                "type": "Role",
                "principalId": "EXAMPLEROLE",
                "arn": "arn:AWS:iam::111122223333:role/Sampleuser01",
                "accountId": "111122223333",
                "userName": "Sampleuser01"
            },
            "webIdFederationData": {},
            "attributes": {
                "creationDate": "2021-06-30T21:17:06Z",
                "mfaAuthenticated": "false"
            }
        },
        "invokedBy": "omics.amazonAWS.com"
    },
    "eventTime": "2021-06-30T21:17:37Z",
    "eventSource": "kms.amazonAWS.com",
    "eventName": "GenerateDataKey",
    "AWSRegion": "us-east-1",
    "sourceIPAddress": "omics.amazonAWS.com",
    "userAgent": "omics.amazonAWS.com",
    "requestParameters": {
```

```
        "keySpec": "AES_256",
        "keyId": "arn:AWS:kms:us-east-1:111122223333:key/EXAMPLE_KEY_ARN"
    },
    "responseElements": null,
    "requestID": "EXAMPLE_ID_01",
    "eventID": "EXAMPLE_ID_02",
    "readOnly": true,
    "resources": [
        {
            "accountId": "111122223333",
            "type": "AWS::KMS::Key",
            "ARN": "arn:AWS:kms:us-east-1:111122223333:key/EXAMPLE_KEY_ARN"
        }
    ],
    "eventType": "AWSApiCall",
    "managementEvent": true,
    "recipientAccountId": "111122223333",
    "eventCategory": "Management"
}
```

Ulteriori informazioni

Le seguenti risorse forniscono ulteriori informazioni sulla crittografia dei dati a riposo.

Per ulteriori informazioni sui concetti di base di AWS Key Management Service, consulta la AWS KMS documentazione.

Per ulteriori informazioni sulle best practice di sicurezza, consulta la AWS KMS documentazione.

Crittografia dei dati in transito

AWS HealthOmics utilizza TLS 1.2+ per crittografare i dati in transito attraverso gli endpoint pubblici e tramite i servizi di backend.

Gestione delle identità e degli accessi in HealthOmics

AWS Identity and Access Management (IAM) è uno strumento Servizio AWS che aiuta un amministratore a controllare in modo sicuro l'accesso alle risorse. AWS Gli amministratori IAM controllano chi può essere autenticato (effettuato l'accesso) e autorizzato (disporre delle autorizzazioni) a utilizzare le risorse AWS. HealthOmics IAM è uno Servizio AWS strumento che puoi utilizzare senza costi aggiuntivi.

Argomenti

Destinatari

Il modo in cui utilizzi AWS Identity and Access Management (IAM) varia in base al tuo ruolo:

- Utente del servizio: richiedi le autorizzazioni all'amministratore se non riesci ad accedere alle funzionalità (consulta Risoluzione dei problemi di AWS HealthOmics identità e accesso)
- Amministratore del servizio: determina l'accesso degli utenti e invia le richieste di autorizzazione (consulta Come AWS HealthOmics funziona con IAM)
- Amministratore IAM: scrivi policy per gestire l'accesso (consulta Esempi di policy basate sull'identità per AWS HealthOmics)

Autenticazione con identità

L'autenticazione è il modo in cui accedi AWS utilizzando le tue credenziali di identità. Devi autenticarti come utente IAM o assumendo un ruolo IAM. Utente root dell'account AWS

Puoi accedere come identità federata utilizzando credenziali provenienti da una fonte di identità come AWS IAM Identity Center (IAM Identity Center), autenticazione Single Sign-On o credenziali. Google/ Facebook Per ulteriori informazioni sull'accesso, consulta Come accedere all' Account AWS nella Guida per l'utente di Accedi ad AWS .

Per l'accesso programmatico, AWS fornisce un SDK e una CLI per firmare crittograficamente le richieste. Per ulteriori informazioni, consulta AWS Signature Version 4 per le richieste API nella Guida per l'utente IAM.

Account AWS utente root

Quando si crea un Account AWS, si inizia con un'identità di accesso denominata utente Account AWS root che ha accesso completo a tutte Servizi AWS le risorse. Si consiglia vivamente di non utilizzare l'utente root per le attività quotidiane. Per le attività che richiedono le credenziali dell'utente root, consulta Attività che richiedono le credenziali dell'utente root nella Guida per l'utente IAM.

Identità federata

Come procedura ottimale, richiedi agli utenti umani di utilizzare la federazione con un provider di identità per accedere Servizi AWS utilizzando credenziali temporanee.

Un'identità federata è un utente della directory aziendale, del provider di identità Web o Directory Service che accede Servizi AWS utilizzando le credenziali di una fonte di identità. Le identità federate assumono ruoli che forniscono credenziali temporanee.

Per la gestione centralizzata degli accessi, si consiglia di utilizzare AWS IAM Identity Center. Per ulteriori informazioni, consulta Che cos'è il Centro identità IAM? nella Guida per l'utente di AWS IAM Identity Center .

Utenti e gruppi IAM

Un utente IAM è una identità che dispone di autorizzazioni specifiche per una singola persona o applicazione. Consigliamo di utilizzare credenziali temporanee invece di utenti IAM con credenziali a lungo termine. Per ulteriori informazioni, consulta Richiedere agli utenti umani di utilizzare la federazione con un provider di identità per accedere AWS utilizzando credenziali temporanee nella Guida per l'utente IAM.

Un gruppo IAM specifica una raccolta di utenti IAM e semplifica la gestione delle autorizzazioni per gestire gruppi di utenti di grandi dimensioni. Per ulteriori informazioni, consulta Casi d'uso per utenti IAM nella Guida per l'utente di IAM.

Ruoli IAM

Un ruolo IAM è un'identità con autorizzazioni specifiche che fornisce credenziali temporanee. Puoi assumere un ruolo passando da un ruolo utente a un ruolo IAM (console) o chiamando un'operazione AWS CLI o AWS API. Per ulteriori informazioni, consulta Metodi per assumere un ruolo nella Guida per l'utente IAM.

I ruoli IAM sono utili per l'accesso federato degli utenti, le autorizzazioni utente IAM temporanee, l'accesso tra account, l'accesso tra servizi e le applicazioni in esecuzione su Amazon. EC2 Per

maggiori informazioni, consultare Accesso a risorse multi-account in IAM nella Guida per l'utente IAM.

Gestione dell'accesso tramite policy

Puoi controllare l'accesso AWS creando policy e collegandole a identità o risorse. AWS Una policy definisce le autorizzazioni quando è associata a un'identità o a una risorsa. AWS valuta queste politiche quando un preside effettua una richiesta. La maggior parte delle politiche viene archiviata AWS come documenti JSON. Per maggiori informazioni sui documenti delle policy JSON, consulta Panoramica delle policy JSON nella Guida per l'utente IAM.

Utilizzando le policy, gli amministratori specificano chi ha accesso a cosa definendo quale principale può eseguire azioni su quali risorse e in quali condizioni.

Per impostazione predefinita, utenti e ruoli non dispongono di autorizzazioni. Un amministratore IAM crea le policy IAM e le aggiunge ai ruoli, che gli utenti possono quindi assumere. Le policy IAM definiscono le autorizzazioni indipendentemente dal metodo utilizzato per eseguirle.

Policy basate sull'identità

Le policy basate su identità sono documenti di policy di autorizzazione JSON che è possibile collegare a un'identità (utente, gruppo o ruolo). Tali policy controllano le operazioni autorizzate per l'identità, nonché le risorse e le condizioni in cui possono essere eseguite. Per informazioni su come creare una policy basata su identità, consultare Definizione di autorizzazioni personalizzate IAM con policy gestite dal cliente nella Guida per l'utente IAM.

Le policy basate su identità possono essere policy in linea (con embedding direttamente in una singola identità) o policy gestite (policy autonome collegate a più identità). Per informazioni su come scegliere tra una policy gestita o una policy inline, consulta Scegliere tra policy gestite e policy in linea nella Guida per l'utente di IAM.

Policy basate sulle risorse

Le policy basate su risorse sono documenti di policy JSON che è possibile collegare a una risorsa. Gli esempi includono le policy di trust dei ruoli IAM e le policy dei bucket di Amazon S3. Nei servizi che supportano policy basate sulle risorse, gli amministratori dei servizi possono utilizzarli per controllare l'accesso a una risorsa specifica. In una policy basata sulle risorse è obbligatorio specificare un'entità principale.

Le policy basate sulle risorse sono policy inline che si trovano in tale servizio. Non è possibile utilizzare le policy AWS gestite di IAM in una policy basata sulle risorse.

Altri tipi di policy

AWS supporta tipi di policy aggiuntivi che possono impostare le autorizzazioni massime concesse dai tipi di policy più comuni:

- Limiti delle autorizzazioni: imposta il numero massimo di autorizzazioni che una policy basata su identità ha la possibilità di concedere a un'entità IAM. Per ulteriori informazioni, consulta Limiti delle autorizzazioni per le entità IAM nella Guida per l'utente di IAM.
- Politiche di controllo del servizio (SCPs): specificano le autorizzazioni massime per un'organizzazione o un'unità organizzativa in. AWS Organizations Per ulteriori informazioni, consultare Policy di controllo dei servizi nella Guida per l'utente di AWS Organizations .
- Politiche di controllo delle risorse (RCPs): imposta le autorizzazioni massime disponibili per le risorse nei tuoi account. Per ulteriori informazioni, consulta Politiche di controllo delle risorse (RCPs) nella Guida per l'AWS Organizations utente.
- Policy di sessione: policy avanzate passate come parametro quando si crea una sessione temporanea per un ruolo o un utente federato. Per maggiori informazioni, consultare Policy di sessione nella Guida per l'utente IAM.

Più tipi di policy

Quando a una richiesta si applicano più tipi di policy, le autorizzazioni risultanti sono più complicate da comprendere. Per scoprire come si AWS determina se consentire o meno una richiesta quando sono coinvolti più tipi di policy, consulta Logica di valutazione delle policy nella IAM User Guide.

Come AWS HealthOmics funziona con IAM

Prima di utilizzare IAM per gestire l'accesso ad AWS HealthOmics, scopri quali funzionalità IAM sono disponibili per l'uso con AWS HealthOmics.

Funzionalità IAM che puoi utilizzare con AWS HealthOmics

Funzionalità IAM	HealthOmics supporto
Policy basate sull'identità	Sì

Funzionalità IAM	HealthOmics supporto
Policy basate su risorse	No
Operazioni di policy	Sì
Risorse relative alle policy	Sì
Chiavi di condizione delle policy	No
ACLs	No
ABAC (tag nelle politiche)	Sì
Credenziali temporanee	Sì
Autorizzazioni del principale	Sì
Ruoli di servizio	Sì
Ruoli collegati al servizio	No

Per avere una visione di alto livello di come HealthOmics e altri AWS servizi funzionano con la maggior parte delle funzionalità IAM, consulta AWS i servizi che funzionano con IAM nella IAM User Guide.

Prevenzione del problema "confused deputy" tra servizi

Il problema confused deputy è un problema di sicurezza in cui un'entità che non dispone dell'autorizzazione per eseguire un'azione può costringere un'entità maggiormente privilegiata a eseguire l'azione. Nel AWS, l'impersonificazione tra servizi può portare al problema del vicesceriffo confuso. La rappresentazione tra servizi può verificarsi quando un servizio (il servizio chiamante) effettua una chiamata a un altro servizio (il servizio chiamato). Il servizio chiamante può essere manipolato per utilizzare le proprie autorizzazioni e agire sulle risorse di un altro cliente, a cui normalmente non avrebbe accesso. Per evitare ciò, AWS fornisce alcuni strumenti che consentono di proteggere i dati per tutti i servizi che dispongono di principali del servizio a cui è stato consentito l'accesso alle risorse del tuo account.

Consigliamo di utilizzare le chiavi di contesto `aws:SourceArn` e `aws:SourceAccount` global condition nelle policy delle risorse per limitare le autorizzazioni che AWS HealthOmics concede a un altro servizio alla risorsa.

Per evitare il problema confuso dei vicedirettori nei ruoli assunti da HealthOmics, imposta il valore di `aws:SourceArn` to `arn:aws:omics:region:accountNumber:*` nella politica di fiducia del ruolo. La wildcard (*) applica la condizione a tutte le HealthOmics risorse.

La seguente politica sulle relazioni di fiducia concede HealthOmics l'accesso alle risorse e utilizza le chiavi di contesto della `aws:SourceArn` condizione `aws:SourceAccount` globale per prevenire il confuso problema del vicesceriffo. Utilizza questa politica quando crei un ruolo per HealthOmics.

JSON

```
{
  "Version":"2012-10-17",
  "Statement": [
    {
      "Sid": "",
      "Effect": "Allow",
      "Principal": {
        "Service": [
          "omics.amazonaws.com"
        ]
      },
      "Action": "sts:AssumeRole",
      "Condition": {
        "StringEquals": {
          "aws:SourceAccount": "123456789012"
        },
        "ArnLike": {
          "aws:SourceArn": "arn:aws:omics:us-east-1:123456789012:*"
        }
      }
    }
  ]
}
```

Politiche basate sull'identità per HealthOmics

Supporta le policy basate sull'identità: sì

Le policy basate sull'identità sono documenti di policy di autorizzazione JSON che è possibile allegare a un'identità (utente, gruppo di utenti o ruolo IAM). Tali policy definiscono le operazioni che utenti e ruoli possono eseguire, su quali risorse e in quali condizioni. Per informazioni su come creare una policy basata su identità, consulta Definizione di autorizzazioni personalizzate IAM con policy gestite dal cliente nella Guida per l'utente di IAM.

Con le policy basate sull'identità di IAM, è possibile specificare quali operazioni e risorse sono consentite o respinte, nonché le condizioni in base alle quali le operazioni sono consentite o respinte. Per informazioni su tutti gli elementi utilizzabili in una policy JSON, consulta Guida di riferimento agli elementi delle policy JSON IAM nella Guida per l'utente IAM.

Esempi di politiche basate sull'identità per HealthOmics

Per visualizzare esempi di policy HealthOmics basate sull'identità di AWS, consulta. Esempi di policy basate sull'identità per AWS HealthOmics

Politiche basate sulle risorse all'interno HealthOmics

Supporta le policy basate su risorse: no

Le policy basate su risorse sono documenti di policy JSON che è possibile collegare a una risorsa. Esempi di policy basate sulle risorse sono le policy di attendibilità dei ruoli IAM e le policy di bucket Amazon S3. Nei servizi che supportano policy basate sulle risorse, gli amministratori dei servizi possono utilizzarli per controllare l'accesso a una risorsa specifica. Quando è collegata a una risorsa, una policy definisce le operazioni che un principale può eseguire su tale risorsa e a quali condizioni. In una policy basata sulle risorse è obbligatorio specificare un'entità principale. I principali possono includere account, utenti, ruoli, utenti federati o. Servizi AWS

Per consentire l'accesso multi-account, è possibile specificare un intero account o entità IAM in un altro account come entità principale in una policy basata sulle risorse. Per ulteriori informazioni, consulta Accesso a risorse multi-account in IAM nella Guida per l'utente IAM.

Azioni politiche per HealthOmics

Supporta le operazioni di policy: si

Gli amministratori possono utilizzare le policy AWS JSON per specificare chi ha accesso a cosa. In altre parole, quale entità principale può eseguire operazioni su quali risorse e in quali condizioni.

L'elemento `Action` di una policy JSON descrive le operazioni che è possibile utilizzare per consentire o negare l'accesso in una policy. Includere le operazioni in una policy per concedere le autorizzazioni a eseguire l'operazione associata.

Per visualizzare un elenco di HealthOmics azioni, consulta Actions Defined by AWS HealthOmics nel Service Authorization Reference.

Le azioni politiche in HealthOmics uso utilizzano il seguente prefisso prima dell'azione:

```
omics
```

Per specificare più operazioni in una sola istruzione, occorre separarle con la virgola.

```
"Action": [
      "omics:action1",
      "omics:action2"
         ]
```

Per visualizzare esempi di policy HealthOmics basate sull'identità di AWS, consulta. Esempi di policy basate sull'identità per AWS HealthOmics

Risorse politiche per HealthOmics

Supporta le risorse relative alle policy: sì

Gli amministratori possono utilizzare le policy AWS JSON per specificare chi ha accesso a cosa. In altre parole, quale entità principale può eseguire operazioni su quali risorse e in quali condizioni.

L'elemento JSON `Resource` della policy specifica l'oggetto o gli oggetti ai quali si applica l'operazione. Come best practice, specifica una risorsa utilizzando il suo nome della risorsa Amazon (ARN). Per le azioni che non supportano le autorizzazioni a livello di risorsa, si utilizza un carattere jolly (*) per indicare che l'istruzione si applica a tutte le risorse.

```
"Resource": "*"
```

Per visualizzare un elenco dei tipi di HealthOmics risorse e relativi ARNs, consulta Resources Defined by AWS HealthOmics nel Service Authorization Reference. Per sapere con quali azioni è possibile specificare l'ARN di ogni risorsa, consulta Actions Defined by AWS. HealthOmics

Per visualizzare esempi di policy HealthOmics basate sull'identità di AWS, consulta. Esempi di policy basate sull'identità per AWS HealthOmics

Chiavi relative alle condizioni della policy per HealthOmics

Le chiavi delle condizioni delle politiche non sono supportate in HealthOmics.

Liste di controllo degli accessi (ACLs) in HealthOmics

Supporti ACLs: no

Le liste di controllo degli accessi (ACLs) controllano quali principali (membri dell'account, utenti o ruoli) dispongono delle autorizzazioni per accedere a una risorsa. ACLs sono simili alle politiche basate sulle risorse, sebbene non utilizzino il formato del documento di policy JSON.

Controllo degli accessi basato sugli attributi (ABAC) con HealthOmics

Supporta ABAC (tag nelle policy): sì

Il controllo degli accessi basato su attributi (ABAC) è una strategia di autorizzazione che definisce le autorizzazioni in base ad attributi chiamati tag. Puoi allegare tag a entità e AWS risorse IAM, quindi progettare politiche ABAC per consentire le operazioni quando il tag del principale corrisponde al tag sulla risorsa.

Per controllare l'accesso basato su tag, fornire informazioni sui tag nell'elemento condizione di una policy utilizzando le chiavi di condizione `aws:ResourceTag/key-name`, `aws:RequestTag/key-name` o `aws:TagKeys`.

Se un servizio supporta tutte e tre le chiavi di condizione per ogni tipo di risorsa, il valore per il servizio è Sì. Se un servizio supporta tutte e tre le chiavi di condizione solo per alcuni tipi di risorsa, allora il valore sarà Parziale.

Per maggiori informazioni su ABAC, consulta Definizione delle autorizzazioni con autorizzazione ABAC nella Guida per l'utente di IAM. Per visualizzare un tutorial con i passaggi per l'impostazione di

ABAC, consulta Utilizzo del controllo degli accessi basato su attributi (ABAC) nella Guida per l'utente di IAM.

Per ulteriori informazioni sul tagging delle risorse di HealthOmics, consulta Taggare le risorse in HealthOmics.

L'esempio seguente mostra come scrivere una policy IAM che neghi l'accesso a una risorsa senza un tag specifico.

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Effect": "Deny",
            "Action": [
                "omics:*"
            ],
            "Resource": [
                "*"
            ],
            "Condition": {
                "Null": {
                  "aws:RequestTag/MyCustomTag": "true"
                }
            }
        }
    ]
}
```

Utilizzo di credenziali temporanee con HealthOmics

Supporta le credenziali temporanee: sì

Le credenziali temporanee forniscono l'accesso a breve termine alle AWS risorse e vengono create automaticamente quando si utilizza la federazione o si cambia ruolo. AWS consiglia di generare dinamicamente credenziali temporanee anziché utilizzare chiavi di accesso a lungo termine. Per

ulteriori informazioni, consulta Credenziali di sicurezza temporanee in IAM e Servizi AWS compatibili con IAM nella Guida per l'utente IAM.

Autorizzazioni principali multiservizio per HealthOmics

Supporta l'inoltro delle sessioni di accesso (FAS): sì

Le sessioni di accesso inoltrato (FAS) utilizzano le autorizzazioni del principale che chiama an Servizio AWS, combinate con la richiesta di effettuare richieste Servizio AWS ai servizi downstream. Per i dettagli delle policy relative alle richieste FAS, consulta la pagina Inoltro sessioni di accesso.

Ruoli di servizio per HealthOmics

Supporta i ruoli di servizio: sì

Un ruolo di servizio è un ruolo IAM che un servizio assume per eseguire operazioni per tuo conto. Un amministratore IAM può creare, modificare ed eliminare un ruolo di servizio dall'interno di IAM. Per ulteriori informazioni, consulta Create a role to delegate permissions to an Servizio AWS nella Guida per l'utente IAM.

> **Warning**
>
> La modifica delle autorizzazioni per un ruolo di servizio potrebbe compromettere la funzionalità. HealthOmics Modifica i ruoli di servizio solo quando viene HealthOmics fornita una guida in tal senso.

Ruoli collegati ai servizi per HealthOmics

Supporta i ruoli collegati ai servizi: no

Un ruolo collegato al servizio è un tipo di ruolo di servizio collegato a un. Servizio AWS Il servizio può assumere il ruolo per eseguire un'operazione per tuo conto. I ruoli collegati al servizio vengono visualizzati nel tuo account Account AWS e sono di proprietà del servizio. Un amministratore IAM può visualizzare le autorizzazioni per i ruoli collegati al servizio, ma non modificarle.

Per ulteriori informazioni su come creare e gestire i ruoli collegati ai servizi, consulta Servizi AWS supportati da IAM. Trova un servizio nella tabella che include un `Yes` nella colonna Service-linked role (Ruolo collegato ai servizi). Scegli il collegamento Sì per visualizzare la documentazione relativa al ruolo collegato ai servizi per tale servizio.

Esempi di policy basate sull'identità per AWS HealthOmics

Per impostazione predefinita, gli utenti e i ruoli non dispongono dell'autorizzazione per creare o modificare HealthOmics risorse AWS. Per concedere agli utenti l'autorizzazione a eseguire azioni sulle risorse di cui hanno bisogno, un amministratore IAM può creare policy IAM.

Per informazioni su come creare una policy basata su identità IAM utilizzando questi documenti di policy JSON di esempio, consulta Creazione di policy IAM (console) nella Guida per l'utente di IAM.

Per dettagli sulle azioni e sui tipi di risorse definiti da AWS HealthOmics, incluso il formato di ARNs per ogni tipo di risorsa, consulta Actions, Resources and Condition Keys for AWS HealthOmics nel Service Authorization Reference.

Argomenti

Best practice per le policy

Le policy basate sull'identità determinano se qualcuno può creare, accedere o eliminare HealthOmics risorse AWS nel tuo account. Queste azioni possono comportare costi aggiuntivi per l' Account AWS. Quando si creano o modificano policy basate sull'identità, seguire queste linee guida e raccomandazioni:

- Inizia con le policy AWS gestite e passa alle autorizzazioni con privilegi minimi: per iniziare a concedere autorizzazioni a utenti e carichi di lavoro, utilizza le policy gestite che concedono le autorizzazioni per molti casi d'uso comuni.AWS Sono disponibili nel tuo. Account AWS Ti consigliamo di ridurre ulteriormente le autorizzazioni definendo politiche gestite dai AWS clienti specifiche per i tuoi casi d'uso. Per maggiori informazioni, consulta Policy gestite da AWS o Policy gestite da AWS per le funzioni dei processi nella Guida per l'utente di IAM.
- Applicazione delle autorizzazioni con privilegio minimo - Quando si impostano le autorizzazioni con le policy IAM, concedere solo le autorizzazioni richieste per eseguire un'attività. È possibile farlo definendo le azioni che possono essere intraprese su risorse specifiche in condizioni specifiche, note anche come autorizzazioni con privilegio minimo. Per maggiori informazioni sull'utilizzo di IAM per applicare le autorizzazioni, consulta Policy e autorizzazioni in IAM nella Guida per l'utente di IAM.

- Condizioni d'uso nelle policy IAM per limitare ulteriormente l'accesso - Per limitare l'accesso ad azioni e risorse è possibile aggiungere una condizione alle policy. Ad esempio, è possibile scrivere una condizione di policy per specificare che tutte le richieste devono essere inviate utilizzando SSL. Puoi anche utilizzare le condizioni per concedere l'accesso alle azioni del servizio se vengono utilizzate tramite uno specifico Servizio AWS, ad esempio CloudFormation. Per maggiori informazioni, consultare la sezione Elementi delle policy JSON di IAM: condizione nella Guida per l'utente di IAM.
- Utilizzo dello strumento di analisi degli accessi IAM per convalidare le policy IAM e garantire autorizzazioni sicure e funzionali - Lo strumento di analisi degli accessi IAM convalida le policy nuove ed esistenti in modo che aderiscano al linguaggio (JSON) della policy IAM e alle best practice di IAM. Lo strumento di analisi degli accessi IAM offre oltre 100 controlli delle policy e consigli utili per creare policy sicure e funzionali. Per maggiori informazioni, consultare Convalida delle policy per il Sistema di analisi degli accessi IAM nella Guida per l'utente di IAM.
- Richiedi l'autenticazione a più fattori (MFA): se hai uno scenario che richiede utenti IAM o un utente root nel Account AWS tuo, attiva l'MFA per una maggiore sicurezza. Per richiedere la MFA quando vengono chiamate le operazioni API, aggiungere le condizioni MFA alle policy. Per maggiori informazioni, consultare Protezione dell'accesso API con MFA nella Guida per l'utente di IAM.

Per maggiori informazioni sulle best practice in IAM, consultare Best practice di sicurezza in IAM nella Guida per l'utente IAM.

Utilizzo della console di HealthOmics

Per accedere alla HealthOmics console AWS, devi disporre di un set minimo di autorizzazioni. Queste autorizzazioni devono consentirti di elencare e visualizzare i dettagli sulle HealthOmics risorse AWS presenti nel tuo Account AWS. Se crei una policy basata sull'identità più restrittiva rispetto alle autorizzazioni minime richieste, la console non funzionerà nel modo previsto per le entità (utenti o ruoli) associate a tale policy.

Non è necessario concedere autorizzazioni minime per la console agli utenti che effettuano chiamate solo verso AWS CLI o l' AWS API. Al contrario, è opportuno concedere l'accesso solo alle azioni che corrispondono all'operazione API che stanno cercando di eseguire.

Consentire agli utenti di visualizzare le loro autorizzazioni

Questo esempio mostra in che modo è possibile creare una policy che consente agli utenti IAM di visualizzare le policy inline e gestite che sono collegate alla relativa identità utente. Questa politica

include le autorizzazioni per completare questa azione sulla console o utilizzando l'API o a livello di codice. AWS CLI AWS

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "ViewOwnUserInfo",
            "Effect": "Allow",
            "Action": [
                "iam:GetUserPolicy",
                "iam:ListGroupsForUser",
                "iam:ListAttachedUserPolicies",
                "iam:ListUserPolicies",
                "iam:GetUser"
            ],
            "Resource": ["arn:aws:iam::*:user/${aws:username}"]
        },
        {
            "Sid": "NavigateInConsole",
            "Effect": "Allow",
            "Action": [
                "iam:GetGroupPolicy",
                "iam:GetPolicyVersion",
                "iam:GetPolicy",
                "iam:ListAttachedGroupPolicies",
                "iam:ListGroupPolicies",
                "iam:ListPolicyVersions",
                "iam:ListPolicies",
                "iam:ListUsers"
            ],
            "Resource": "*"
        }
    ]
}
```

AWS politiche gestite per AWS HealthOmics

Una politica AWS gestita è una politica autonoma creata e amministrata da AWS. AWS le politiche gestite sono progettate per fornire autorizzazioni per molti casi d'uso comuni, in modo da poter iniziare ad assegnare autorizzazioni a utenti, gruppi e ruoli.

Tieni presente che le policy AWS gestite potrebbero non concedere le autorizzazioni con il privilegio minimo per i tuoi casi d'uso specifici, poiché sono disponibili per tutti i clienti. AWS Si consiglia pertanto di ridurre ulteriormente le autorizzazioni definendo policy gestite dal cliente specifiche per i propri casi d'uso.

Non è possibile modificare le autorizzazioni definite nelle politiche gestite. AWS Se AWS aggiorna le autorizzazioni definite in una politica AWS gestita, l'aggiornamento ha effetto su tutte le identità principali (utenti, gruppi e ruoli) a cui è associata la politica. AWS è più probabile che aggiorni una policy AWS gestita quando ne Servizio AWS viene lanciata una nuova o quando diventano disponibili nuove operazioni API per i servizi esistenti.

Per ulteriori informazioni, consultare Policy gestite da AWS nella Guida per l'utente di IAM.

AWS politica gestita: AmazonOmicsFullAccess

Puoi allegare la `AmazonOmicsFullAccess` policy alle tue identità IAM per dare loro pieno accesso a HealthOmics.

Questa policy concede le autorizzazioni di accesso complete a tutte le azioni. HealthOmics Quando crei un'annotazione o un archivio di varianti, Omics ti consentirà anche di accedere a quel negozio tramite un Resource Share Invitation nella console Resource Access Manager (RAM). Per ulteriori informazioni sugli inviti alla condivisione delle risorse tramite Lake Formation, consulta la sezione Condivisione dei dati tra account in Lake Formation. Per una politica di amministrazione di Omics, sono necessarie anche le seguenti autorizzazioni per accedere al tuo bucket Amazon S3.

- PutObject
- GetObject
- ListBucket
- AbortMultipartUpload
- ListMultipartUploadParts

JSON

```
{
 "Version":"2012-10-17",
 "Statement": [
  {
   "Effect": "Allow",
   "Action": [
    "omics:*"
   ],
   "Resource": "*"
  },
  {
   "Effect": "Allow",
   "Action": [
    "ram:AcceptResourceShareInvitation",
    "ram:GetResourceShareInvitations"
   ],
   "Resource": "*",
   "Condition": {
    "StringEquals": {
     "aws:CalledViaLast": "omics.amazonaws.com"
    }
   }
  },
  {
   "Effect": "Allow",
   "Action": "iam:PassRole",
   "Resource": "*",
   "Condition": {
    "StringEquals": {
     "iam:PassedToService": "omics.amazonaws.com"
    }
   }
  }
 ]
}
```

AWS politica gestita: AmazonOmicsReadOnlyAccess

Puoi allegare la `AWSOmicsReadOnlyAccess` policy alle tue identità IAM quando desideri limitare le autorizzazioni di tale identità all'accesso in sola lettura.

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "omics:Get*",
                "omics:List*"
            ],
            "Resource": "*"
        }
    ]
}
```

HealthOmics aggiornamenti alle politiche gestite AWS

Visualizza i dettagli sugli aggiornamenti delle politiche AWS gestite HealthOmics da quando questo servizio ha iniziato a tenere traccia di queste modifiche. Per ricevere avvisi automatici sulle modifiche a questa pagina, iscriviti al feed RSS nella pagina della cronologia dei HealthOmics documenti.

Modifica	Descrizione	Data
AmazonOmicsFullAccess - Aggiunta una nuova politica	HealthOmics ha aggiunto una nuova politica per garantire a un utente l'accesso completo a tutte le azioni e le risorse. Per ulteriori informazioni, consulta AmazonOmicsFullAccess.	23 febbraio 2023

Modifica	Descrizione	Data
HealthOmics ha iniziato a tenere traccia delle modifiche	HealthOmics ha iniziato a tenere traccia delle modifiche per le sue politiche AWS gestite.	29 novembre 2022
AmazonOmicsReadOnlyAccess - Aggiunta una nuova politica	HealthOmics ha aggiunto una nuova politica che limita l'accesso alla sola lettura. Per ulteriori informazioni, vedi AmazonOmicsReadOnlyAccess.	29 novembre 2022

Risoluzione dei problemi di AWS HealthOmics identità e accesso

Utilizza le seguenti informazioni per aiutarti a diagnosticare e risolvere i problemi più comuni che potresti riscontrare quando lavori con AWS HealthOmics e IAM.

Argomenti

- Non sono autorizzato a eseguire alcuna azione in HealthOmics
- Non sono autorizzato a eseguire iam: PassRole
- Voglio consentire a persone esterne a me di accedere Account AWS alle mie HealthOmics risorse

Non sono autorizzato a eseguire alcuna azione in HealthOmics

Se ricevi un errore che indica che non sei autorizzato a eseguire un'operazione, le tue policy devono essere aggiornate per poter eseguire l'operazione.

L'errore di esempio seguente si verifica quando l'utente IAM `mateojackson` prova a utilizzare la console per visualizzare i dettagli relativi a una risorsa *`my-example-widget`* fittizia ma non dispone di autorizzazioni `omics:`*`GetWidget`* fittizie.

```
User: arn:aws:iam::123456789012:user/mateojackson is not authorized to perform:
 omics:GetWidget on resource: my-example-widget
```

In questo caso, la policy per l'utente `mateojackson` deve essere aggiornata per consentire l'accesso alla risorsa *`my-example-widget`* utilizzando l'azione `omics:`*`GetWidget`*.

Se hai bisogno di aiuto, contatta il tuo AWS amministratore. L'amministratore è la persona che ti ha fornito le credenziali di accesso.

Non sono autorizzato a eseguire iam: PassRole

Se ricevi un messaggio di errore indicante che non sei autorizzato a eseguire l'`iam:PassRole`azione, le tue policy devono essere aggiornate per consentirti di trasferire un ruolo ad AWS HealthOmics.

Alcuni Servizi AWS consentono di trasferire un ruolo esistente a quel servizio invece di creare un nuovo ruolo di servizio o un ruolo collegato al servizio. Per eseguire questa operazione, è necessario disporre delle autorizzazioni per trasmettere il ruolo al servizio.

Il seguente errore di esempio si verifica quando un utente IAM denominato `marymajor` tenta di utilizzare la console per eseguire un'azione in AWS HealthOmics. Tuttavia, l'azione richiede che il servizio disponga delle autorizzazioni concesse da un ruolo di servizio. Mary non dispone delle autorizzazioni per trasmettere il ruolo al servizio.

```
User: arn:aws:iam::123456789012:user/marymajor is not authorized to perform:
 iam:PassRole
```

In questo caso, le policy di Mary devono essere aggiornate per poter eseguire l'operazione `iam:PassRole`.

Se hai bisogno di aiuto, contatta il tuo AWS amministratore. L'amministratore è la persona che ti ha fornito le credenziali di accesso.

Voglio consentire a persone esterne a me di accedere Account AWS alle mie HealthOmics risorse

È possibile creare un ruolo con il quale utenti in altri account o persone esterne all'organizzazione possono accedere alle tue risorse. È possibile specificare chi è attendibile per l'assunzione del ruolo. Per i servizi che supportano politiche basate sulle risorse o liste di controllo degli accessi (ACLs), puoi utilizzare tali politiche per concedere alle persone l'accesso alle tue risorse.

Per maggiori informazioni, consulta gli argomenti seguenti:

- Per sapere se AWS HealthOmics supporta queste funzionalità, consultaCome AWS HealthOmics funziona con IAM.
- Per scoprire come fornire l'accesso alle tue risorse attraverso Account AWS le risorse di tua proprietà, consulta Fornire l'accesso a un utente IAM in un altro Account AWS di tua proprietà nella IAM User Guide.
- Per scoprire come fornire l'accesso alle tue risorse a terze parti Account AWS, consulta Fornire l'accesso a soggetti Account AWS di proprietà di terze parti nella Guida per l'utente IAM.
- Per informazioni su come fornire l'accesso tramite la federazione delle identità, consulta Fornire l'accesso a utenti autenticati esternamente (federazione delle identità) nella Guida per l'utente IAM.
- Per informazioni sulle differenze di utilizzo tra ruoli e policy basate su risorse per l'accesso multi-account, consulta Accesso a risorse multi-account in IAM nella Guida per l'utente IAM.

Convalida della conformità per AWS HealthOmics

I revisori esterni valutano la sicurezza e la conformità nell' AWS HealthOmics ambito di più programmi di AWS conformità. Ciò include HIPAA, FedRAMP e altri. La tabella seguente mostra le certificazioni di conformità per il servizio. HealthOmics

Certificazione	Link
HIPAA	Riferimento ai servizi idonei alla normativa HIPAA
HiTrust-CSF	Quadro di sicurezza comune della Health Information Trust Alliance
FedRAMP Moderate (Est/Ovest)	Programma federale di gestione dei rischi e delle autorizzazioni
STELLA ISO/CSA	Certificato ISO e CSA STAR
C5	Catalogo dei controlli di conformità del cloud computing
DoD CC SRG IL2	Guida ai requisiti di sicurezza del cloud computing del Dipartimento della Difesa

Certificazione	Link
ENS High	Esquema Nacional de Seguridad
FINMA	Autorità svizzera di vigilanza sui mercati finanziari
È UNA MAPPA	Programma di gestione e valutazione della sicurezza dei sistemi informativi
OSPAR	Rapporto di audit del fornitore di servizi esternalizzato
PCI	Standard di sicurezza dei dati del settore delle carte di pagamento
Pinakes	Associazione bancaria CCI - Third Party Qualification
PiTuKri	Criteri per la valutazione della sicurezza delle informazioni dei servizi cloud
SOC 1,2,3	Controlli di sistema e organizzazione

Per un elenco di tutti i AWS servizi che rientrano nell'ambito di programmi di conformità specifici, consulta Servizi AWS nell'ambito del programma di conformità . Per informazioni generali, consulta Programmi di conformitàAWS.

Puoi scaricare report di audit di terze parti utilizzando AWS Artifact. Per ulteriori informazioni, consulta Scaricamento dei report in AWS Artifact .

HealthOmics gli archivi di dati utilizzano l'ID di esempio per la denominazione interna dei file e per l'etichettatura delle risorse. Prima di importare i dati, controlla se l'ID di esempio contiene dati PHI. In caso affermativo, modificate l'ID del campione prima di importare i dati. Per ulteriori informazioni, consulta le linee guida sulla pagina web sulla conformità AWS HIPAA.

La vostra responsabilità di conformità durante l'utilizzo AWS HealthOmics è determinata dalla sensibilità dei dati, dagli obiettivi di conformità dell'azienda e dalle leggi e dai regolamenti applicabili. AWS fornisce le seguenti risorse per contribuire alla conformità:

- Security and Compliance Quick Start Guides (Guide Quick Start Sicurezza e compliance): queste guide alla distribuzione illustrano considerazioni relative all'architettura e forniscono procedure per la distribuzione di ambienti di base incentrati sulla sicurezza e sulla conformità su AWS.
- Whitepaper sull'architettura per la sicurezza e la conformità HIPAA: questo white paper descrive come le aziende possono utilizzare per creare applicazioni conformi allo standard HIPAA. AWS
- AWS Risorse per la conformità Risorse per la conformità: questa raccolta di potrebbe riguardare il settore e la località in cui operate.
- Evaluating Resources with Rules nella AWS Config Developer Guide: AWS Config valuta la conformità delle configurazioni delle risorse alle pratiche interne, alle linee guida del settore e alle normative.
- AWS Security Hub CSPM— Questo AWS servizio offre una visione completa dello stato di sicurezza dell'utente e consente di verificare la conformità agli standard e alle best practice del settore della sicurezza. AWS

Resilienza in HealthOmics

L'infrastruttura AWS globale è costruita attorno a Regioni AWS zone di disponibilità. Regioni AWS forniscono più zone di disponibilità fisicamente separate e isolate, collegate con reti a bassa latenza, ad alto throughput e altamente ridondanti. Con le zone di disponibilità, puoi progettare e gestire applicazioni e database che eseguono automaticamente il failover tra zone di disponibilità senza interruzioni. Le zone di disponibilità sono più disponibili, tolleranti ai guasti e scalabili rispetto alle infrastrutture a data center singolo o multiplo tradizionali.

Per ulteriori informazioni sulle zone di disponibilità, vedere Global Regioni AWS Infrastructure.AWS

Oltre all'infrastruttura AWS globale, AWS HealthOmics offre diverse funzionalità per supportare le esigenze di resilienza e backup dei dati.

AWS HealthOmics e endpoint VPC di interfaccia ()AWS PrivateLink

Puoi stabilire una connessione privata tra il tuo VPC e creare un AWS HealthOmics endpoint VPC di interfaccia. Gli endpoint di interfaccia sono alimentati da AWS PrivateLink, una tecnologia che puoi utilizzare per accedere in modo privato alle operazioni HealthOmics API senza un gateway Internet, un dispositivo NAT, una connessione VPN o una connessione AWS Direct Connect. Le istanze nel tuo VPC non richiedono indirizzi IP pubblici per comunicare HealthOmics con le operazioni API. Il traffico tra il tuo VPC e HealthOmics non esce dalla rete Amazon.

Ogni endpoint dell'interfaccia è rappresentato da una o più interfacce di rete elastiche nelle tue sottoreti.

Per ulteriori informazioni, consulta Interface VPC endpoints (AWS PrivateLink) nella Amazon VPC User Guide.

Le policy relative agli endpoint VPC sono supportate HealthOmics per tutte le regioni ad eccezione di Israele (Tel Aviv). Per impostazione predefinita, l'accesso completo a HealthOmics è consentito tramite l'endpoint.

Considerazioni sugli endpoint HealthOmics VPC

Prima di configurare un endpoint VPC di interfaccia per HealthOmics, assicurati di esaminare le proprietà e le limitazioni degli endpoint dell'interfaccia nella Amazon VPC User Guide.

HealthOmics supporta l'effettuazione di chiamate a tutte le azioni dell'API di HealthOmics archiviazione dal tuo VPC.

Le policy degli endpoint VPC non sono supportate per impostazione HealthOmics predefinita, ma puoi creare un endpoint VPC per l'accesso completo HealthOmics alle operazioni di storage. HealthOmics Per ulteriori informazioni, consulta Controllo degli accessi ai servizi con endpoint VPC nella Guida per l'utente di Amazon VPC.

Creazione di un endpoint VPC interfaccia per l' HealthOmics

Puoi creare un endpoint VPC per il HealthOmics servizio utilizzando la console Amazon VPC o il (). AWS Command Line Interface AWS CLI Per ulteriori informazioni, consulta Creazione di un endpoint dell'interfaccia nella Guida per l'utente di Amazon VPC.

Crea un endpoint VPC per HealthOmics utilizzando i seguenti nomi di servizio:

- com.amazonaws. *region*.storage-omics
- com.amazonaws. *region*. control-storage-omics
- com.amazonaws. *region*.analisi-omics
- com.amazonaws. *region*.workflows-omics
- com.amazonaws. *region*.tag-omics

Le regioni Stati Uniti orientali (Virginia settentrionale) e Stati Uniti occidentali (Oregon) supportano gli endpoint FIPS. AWS PrivateLink Per queste regioni, puoi anche utilizzare i seguenti nomi di servizio:

- com.amazonaws. *region*. storage-omics-fips
- com.amazonaws. *region*. control-storage-omics-fips
- com.amazonaws. *region*. analytics-omics-fips
- com.amazonaws. *region*. workflows-omics-fips
- com.amazonaws. *region*. tags-omics-fips

Se attivi il DNS privato per l'endpoint, puoi effettuare richieste API a HealthOmics utilizzando il nome DNS predefinito per la regione, ad esempio. `omics.us-east-1.amazonaws.com`

Per ulteriori informazioni, consulta Accesso a un servizio tramite un endpoint dell'interfaccia in Guida per l'utente di Amazon VPC.

Creazione di una policy di endpoint VPC per l' HealthOmics

È possibile allegare un criterio all'endpoint VPC che controlla l'accesso all' HealthOmics. Questa policy specifica le informazioni riportate di seguito:

- Il principale che può eseguire operazioni.
- Le azioni che possono essere eseguite
- Le risorse sui cui si possono eseguire le azioni

Per ulteriori informazioni, consultare Controllo degli accessi ai servizi con endpoint VPC in Guida per l'utente di Amazon VPC.

Esempio: policy degli endpoint VPC per le azioni. HealthOmics

Di seguito è riportato un esempio di policy sugli endpoint per. HealthOmics Se associata a un endpoint, questa policy garantisce l'accesso alle HealthOmics azioni per tutti i principali utenti su tutte le risorse.

API

```
{
   "Statement":[
      {
         "Principal":"*",
         "Effect":"Allow",
         "Action":[
```

```
            "omics:List*"
         ],
         "Resource":"*"
      }
   ]
}
```

AWS CLI

```
aws ec2 modify-vpc-endpoint \
    --vpc-endpoint-id vpce-id \
    --region us-west-2 \
    --policy-document \
    "{\"Statement\":[{\"Principal\":\"*\",\"Effect\":\"Allow\",\"Action\":
[\"omics:List*\"],\"Resource\":\"*\"}]}"
```

Considerazioni speciali per l'accesso ai set di lettura tramite Amazon S3 URIs

Per accedere ai set di lettura tramite Amazon S3 URIs quando utilizzi una connessione privata, configura gli endpoint dell' PrivateLinkinterfaccia nel Sequence Store. Dopo averli configurati, gli endpoint hanno i seguenti formati:

```
com.amazonaws.region.storage-omics
com.amazonaws.region.control-storage-omics
```

Per utilizzare gli endpoint Gateway, segui la guida Gateway endpoints for Amazon S3 per configurare gli endpoint gateway. HealthOmics possiede il bucket Amazon S3, quindi non è necessario creare o modificare la policy del bucket. Gli endpoint Gateway si basano sulla policy associata all'utente o al ruolo che accede ai dati, ma puoi anche configurare gli endpoint con politiche più restrittive. Queste policy possono includere restrizioni all'accesso basate sull'ARN di Amazon S3 Access Point e sulle azioni di Amazon S3.

Monitoraggio di AWS HealthOmics

Il monitoraggio è una parte importante per mantenere l'affidabilità, la disponibilità e le prestazioni di AWS HealthOmics e delle altre AWS soluzioni. AWS fornisce i seguenti strumenti di monitoraggio per monitorare AWS HealthOmics, segnalare quando qualcosa non va e intraprendere azioni automatiche quando appropriato:

- Amazon CloudWatch monitora AWS le tue risorse e le applicazioni su cui esegui AWS in tempo reale. Puoi raccogliere i parametri e tenerne traccia, creare pannelli di controllo personalizzati e impostare allarmi per inviare una notifica o intraprendere azioni quando un parametro specificato raggiunge una determinata soglia. Ad esempio, puoi tenere CloudWatch traccia dell'utilizzo della CPU o di altri parametri delle tue EC2 istanze Amazon e avviare automaticamente nuove istanze quando necessario. Per ulteriori informazioni, consulta la Amazon CloudWatch User Guide.
- Amazon CloudWatch Logs ti consente di monitorare, archiviare e accedere ai tuoi file di registro da EC2 istanze Amazon e altre fonti. CloudTrail CloudWatch I log possono monitorare le informazioni nei file di registro e avvisarti quando vengono raggiunte determinate soglie. Puoi inoltre archiviare i dati del log in storage estremamente durevole. Per ulteriori informazioni, consulta la Amazon CloudWatch Logs User Guide.
- AWS CloudTrail acquisisce le chiamate API e gli eventi correlati effettuati da o per conto del tuo Account AWS e fornisce i file di log a un bucket Simple Storage Service (Amazon S3) specificato. Puoi identificare quali utenti e account hanno richiamato AWS, l'indirizzo IP di origine da cui sono state effettuate le chiamate e quando sono avvenute. Per ulteriori informazioni, consulta la AWS CloudTrail Guida per l'utente di .
- Amazon EventBridge è un servizio di bus eventi senza server che semplifica la connessione delle applicazioni con dati provenienti da una varietà di fonti. EventBridge fornisce un flusso di dati in tempo reale dalle tue applicazioni, applicazioni Software-as-a-Service (SaaS) e AWS servizi e indirizza tali dati verso destinazioni come Lambda. In questo modo puoi monitorare gli eventi che si verificano nei servizi e creare architetture basate su eventi. Per ulteriori informazioni, consulta la Amazon EventBridge User Guide.

> **Note**
>
> Per gli aggiornamenti del servizio, configura e monitora la tua Personal Health Dashboard. Per ulteriori informazioni su come gestire la dashboard, consulta Getting started with your AWS Health Dashboard.

Argomenti

Registrazione degli accessi S3

Puoi monitorare l'accesso dell'API di Amazon S3 ai dati degli archivi in HealthOmics sequenza utilizzando i log di accesso creati dallo store. Puoi utilizzarlo CloudWatch per monitorare l'accesso a S3 dalle operazioni API. HealthOmics CloudWatch offre visibilità sull'accesso ad Amazon S3 proveniente dal tuo account. Se, in qualità di proprietario dei dati, condividi l'accesso a un account di terze parti, la registrazione degli accessi non è disponibile in. CloudWatch Utilizza invece lo S3 Access Log. dello store che registra tutti gli accessi S3 ai dati nel bucket Amazon S3 configurato.

Configura i log di accesso S3 utilizzando le operazioni o API. `CreateSequenceStore` `UpdateSequenceStore` Inoltre, assicurati che il HealthOmics service principal (`omics.amazonaws.com`) disponga delle `s3:PutObject` autorizzazioni per il prefisso S3 configurato.

> **Note**
>
> I log utilizzano la configurazione di crittografia predefinita del bucket di destinazione. Se il bucket utilizza una chiave gestita dal cliente, il responsabile del servizio deve avere accesso per utilizzare la chiave per la scrittura.

Per disattivare la registrazione degli accessi, usa `UpdateSequenceStore` e imposta la configurazione del registro di accesso su vuota.

Monitoraggio HealthOmics con CloudWatch metriche

Puoi monitorare HealthOmics l'utilizzo CloudWatch, che raccoglie dati grezzi e li elabora in metriche leggibili e quasi in tempo reale. Queste statistiche vengono conservate per un periodo di 15 mesi,

per permettere l'accesso alle informazioni storiche e offrire una prospettiva migliore sulle prestazioni del servizio o dell'applicazione web. È anche possibile impostare allarmi che controllano determinate soglie e inviare notifiche o intraprendere azioni quando queste soglie vengono raggiunte. Per ulteriori informazioni, consulta la Amazon CloudWatch User Guide.

Il AWS HealthOmics servizio riporta le seguenti metriche nel `AWS/Omics` namespace.

Le metriche API Call Count sono riportate per quanto segue. AWS HealthOmics APIs Viene riportata solo la dimensione API Operation.

- Riferimento e archivio di riferimento APIs —CreateReferenceStore, DeleteReferenceStore, StartReferenceImportJob
- Set di memorizzazione e lettura delle sequenze APIs — CreateSequenceStore DeleteSequenceStore, StartReadSetImportJob,, StartReadSetActivationJob, StartReadSetExportJob
- Negozio di varianti APIs — CreateVariantStore, DeleteVariantStore, StartVariantImportJob, CancelVariantImportJob
- Archivio annotazioni APIs — CreateAnnotationStore,, DeleteAnotationStore, StartAnnotationImportJob CancelAnnotationImportJob
- Flusso di lavoro, esecuzione ed esecuzione di gruppo APIs — CreateWorkflow, DeleteWorkflow, StartRun,, CancelRun, DeleteRun, CreateRunGroup DeleteRunGroup

Visualizzazione dei parametri *AWS HealthOmics*

CloudWatch le metriche per AWS HealthOmics sono visualizzabili nella console. CloudWatch

Per visualizzare le metriche (console) CloudWatch

1. Accedi alla console di gestione AWS e apri la console CloudWatch .
2. Scegli Metriche, scegli Tutte le metriche, quindi scegli AWS/Usage.
3. Filter Service per. AWS HealthOmics
4. Scegliere la dimensione, selezionare un nome parametro e scegliere Add to graph (Aggiungi a grafico).
5. Seleziona un valore per l'intervallo di date. Il numero di parametri per l'intervallo di date selezionato è visualizzato nel grafico.

Creazione di un allarme utilizzando CloudWatch

Un CloudWatch allarme controlla una singola metrica in un periodo di tempo specificato ed esegue una o più azioni: inviare una notifica a un argomento di Amazon Simple Notification Service (Amazon SNS) o a una politica di Auto Scaling. L'azione o le azioni si basano sul valore della metrica relativo a una determinata soglia in un certo numero di periodi di tempo specificati. CloudWatch può anche inviarti un messaggio Amazon SNS quando l'allarme cambia stato.

CloudWatch gli allarmi richiamano azioni solo quando lo stato cambia e persiste per il periodo specificato.

Per visualizzare le metriche (console) CloudWatch

1. Accedi alla console di gestione AWS e apri la console CloudWatch .
2. Seleziona Alarms (Allarmi), quindi Create Alarm (Crea allarme).
3. Scegli AWS/Usage, quindi scegli una AWS HealthOmics metrica utilizzando la dimensione Servizio.
4. In Time Range (Intervallo di tempo), scegli un intervallo di tempo durante il quale eseguire il monitoraggio, quindi scegli Next (Successivo).
5. Immetti il Name (Nome) e la Description (Descrizione).
6. Per Whenever, scegli >= e digita un valore massimo.
7. Se desideri CloudWatch inviare un'e-mail quando viene raggiunto lo stato di allarme, nella sezione Azioni, per Ogni volta che si verifica un allarme, scegli Lo stato è ALLARME. Per Invia notifica a, scegli una mailing list o scegli Nuova lista e crea una nuova mailing list.
8. Visualizza un'anteprima dell'allarme nella sezione Alarm Preview (Anteprima allarme). Se sei soddisfatto dell'allarme, scegli Create Alarm (Crea allarme).

Monitoraggio HealthOmics con CloudWatch registri

HealthOmics genera una serie di registri per aiutarti a comprendere e risolvere i problemi delle tue esecuzioni. I log sono disponibili in due posizioni: CloudWatch e in Amazon S3.

Per impostazione predefinita, la registrazione delle esecuzioni è attivata. Facoltativamente, puoi disattivare la registrazione di un'esecuzione `LogLevel = OFF` impostando la richiesta. startrun

> **Note**
>
> Per gli aggiornamenti del servizio, configura e monitora la tua Personal Health Dashboard. Per ulteriori informazioni su come gestire la dashboard, consulta Getting started with your AWS Health Dashboard.

Argomenti

Tipi di log per i flussi HealthOmics di lavoro

HealthOmics fornisce i seguenti tipi di log per i flussi di lavoro:

- Registri del motore: i motori di flusso di lavoro sottostanti (Nextflow, WDL e CWL) producono registri del motore per le esecuzioni. Questi registri possono aiutarti a risolvere i problemi di definizione del flusso di lavoro.
- Registri del manifesto di esecuzione: questi registri forniscono informazioni di alto livello su ogni attività in esecuzione, ad esempio lo stato dell'attività, l'ora di inizio, l'ora di fine e il motivo dell'errore (se l'attività non è riuscita).

  I registri del manifesto di esecuzione riportano anche statistiche sull'utilizzo delle risorse che possono essere utili per comprendere le opportunità di ottimizzazione delle risorse. Queste statistiche includono:

  - Media delle CPU
  - CPU massima
  - CPU riservate
  - GPU riservate
  - memoryAverageGiB
  - memoryMaximumGiB

  - memoryReservedGiB
  - Secondi di esecuzione
- Registri di esecuzione: i registri di esecuzione forniscono lo stato di esecuzione generale e l'ora in cui le singole attività vengono avviate, eseguite, interrotte e completate. I registri di esecuzione offrono inoltre visibilità sulle fasi di importazione ed esportazione dei file.
- Registri delle attività: i registri delle attività forniscono informazioni di registrazione dettagliate sulle singole attività eseguite. Gli output nel registro delle attività dipendono dalla definizione dell'attività e dalla posizione in cui vengono utilizzate le istruzioni di registro nel codice. Se i registri delle attività non forniscono il livello di informazioni di cui hai bisogno, prendi in considerazione l'aggiunta di istruzioni di registro aggiuntive alla definizione delle attività per produrre registri delle attività più approfonditi.
- Esegui i registri della cache: i registri della cache di esecuzione forniscono lo stato generale delle cache di esecuzione e della memorizzazione nella cache degli output delle attività. I log della cache di esecuzione offrono visibilità sugli accessi e sugli errori della cache per ogni esecuzione che utilizza la memorizzazione nella cache.
- Outputs.json: per i flussi di lavoro WDL e CWL, HealthOmics fornisce un file generato dal motore, denominato, `outputs.json` al bucket Amazon S3 dopo il completamento dell'esecuzione. Questo file include un elenco e una mappa di tutti gli output per l'esecuzione.

Effettua il login CloudWatch

CloudWatch genera registri del flusso di lavoro per le esecuzioni non riuscite e le esecuzioni riuscite. Tutti i registri sono disponibili per le esecuzioni non riuscite e le esecuzioni riuscite, ad eccezione dei registri del motore, disponibili solo per le esecuzioni non riuscite.

È possibile trovare i registri del CloudWatch flusso di lavoro nel seguente gruppo di registri:. `/aws/omics/WorkflowLog` Inoltre, l'output dell'operazione API get-run fornisce il flusso di registro ARNs per i CloudWatch log del motore e i log di esecuzione.

Per impostazione predefinita, AWS conserva i log a tempo indeterminato. CloudWatch È possibile modificare la politica di conservazione per il gruppo di log in modo da impostare un periodo di conservazione compreso tra 10 anni e un giorno.

La tabella seguente fornisce un riepilogo degli CloudWatch accessi. HealthOmics Tutti i registri del flusso di lavoro sono disponibili per le esecuzioni riuscite e le esecuzioni non riuscite, ad eccezione dei registri del motore, disponibili solo per le esecuzioni non riuscite.

Nome log	Disponibile nei registri CloudWatch	Quando è disponibile il registro	Formato del flusso di registro
Registri del motore	Sì, per le esecuzioni non riuscite	Al termine dell'esecuzione	run/ /engine *runID*
Esegui i registri del manifesto	Sì	Al termine dell'esecuzione	*runID*manifest/ esegui//*runUUID*
Registri di esecuzione	Sì	In tempo reale	esegui/ *runID*
Registri delle attività	Sì	In tempo reale	esegui/ /task/ *runID taskID*
Esegui i log della cache	Sì	In tempo reale	Esegui Cache//*runCacheId runCacheUUID*
Outputs.json (WDL e CWL)	No	N/A	n/a

Accedi ad Amazon S3

Solo i log del motore e il `outputs.json` file vengono consegnati ad Amazon S3.

Al termine di un'esecuzione, i log del motore vengono inviati al bucket S3 e sono disponibili a tempo indeterminato fino a quando non vengono eliminati. Questi log si trovano nella directory logs dell'URI di output S3 specificato per il flusso di lavoro.

Il percorso della directory logs ha il seguente formato:. `s3://{user_provided_path}/logs/`

La tabella seguente fornisce un riepilogo dei HealthOmics log disponibili nel bucket Amazon S3.

Nome log	Disponibile in Amazon S3	Quando è disponibile il registro	Percorso del flusso di log
Registri del motore	Sì	Al termine dell'esecuzione	s3:///logs/engine.log *user_provided_path*
Outputs.json (WDL e CWL)	Sì	Al termine dell'esecuzione	s3://*user_provided_path* /*runID*/logs/outputs.json *runUUID*
Esegui i registri del manifesto, i registri delle esecuzioni e i registri delle attività	No	N/A	n/a

CloudWatch Log interattivi nella CLI

È possibile visualizzare i CloudWatch log in modo interattivo utilizzando il comando Live Tail in modalità interattiva. Puoi monitorare l'avanzamento della corsa in tempo reale e definire fino a 5 parole chiave da evidenziare nei log:

```
aws logs start-live-tail  \
  --mode interactive  \
  --log-group-identifiers arn:aws:logs:region:account-ID:log-group:/aws/omics/
WorkflowLog
```

Per ulteriori informazioni, consulta Start live tail nel AWS CLI Command Reference.

Accesso ai CloudWatch log dalla console

Per accedere ai log di un'esecuzione, puoi collegarti direttamente a questi registri dalla pagina dei dettagli dell'esecuzione nella console. HealthOmics

1. Apri la HealthOmics console.
2. Se necessario, apri il pannello di navigazione a sinistra (≡). Scegli Runs.

3. Seleziona la corsa dalla tabella Runs.
4. Nella pagina dei dettagli dell'esecuzione, puoi scegliere una delle seguenti azioni:

   a. Da Esegui riepilogo, scegli Visualizza registri di esecuzione. La console apre i registri di esecuzione nella CloudWatch console.
   b. Da Esegui riepilogo, scegli Visualizza log in Amazon S3. La console apre la cartella dei log nella console Amazon S3.
   c. Da Esegui attività, scegli Visualizza registri, Visualizza registri di esecuzione o Visualizza registri del manifesto di esecuzione per un'attività. La console apre i registri nella console. CloudWatch

Puoi anche accedere ai log dalla CloudWatch console:

1. Apri la CloudWatch console https://console.aws.amazon.com/cloudwatch/.
2. Dal menu a sinistra, scegli Registra gruppi.
3. Selezionare il gruppo `/aws/omics/WorkflowLog`.

   Se l'elenco dei gruppi di log è lungo, puoi inserire omic nella casella di testo di ricerca per restringere l'elenco.
4. Quando si apre la pagina dei dettagli del gruppo di log, scegli il flusso di log che desideri visualizzare. La console mostra gli eventi per questo flusso di log.

Registrazione delle chiamate AWS HealthOmics API utilizzando AWS CloudTrail

AWS HealthOmics è integrato con AWS CloudTrail, un servizio che fornisce una registrazione delle azioni intraprese da un utente, un ruolo o un AWS servizio in HealthOmics. CloudTrail acquisisce tutte le chiamate API HealthOmics come eventi. Le chiamate acquisite includono chiamate dalla HealthOmics console e chiamate di codice alle operazioni HealthOmics API. Se crei un trail, puoi abilitare la distribuzione continua di CloudTrail eventi a un bucket Amazon S3, inclusi gli eventi per. HealthOmics Se non configuri un percorso, puoi comunque visualizzare gli eventi più recenti nella CloudTrail console nella cronologia degli eventi. Utilizzando le informazioni raccolte da CloudTrail, puoi determinare a quale richiesta è stata inviata HealthOmics, l'indirizzo IP da cui è stata effettuata, chi ha effettuato la richiesta, quando è stata effettuata e dettagli aggiuntivi.

Per ulteriori informazioni CloudTrail, consulta la Guida AWS CloudTrail per l'utente.

HealthOmics informazioni in CloudTrail

CloudTrail è abilitato sul tuo account al Account AWS momento della creazione dell'account. Quando si verifica un'attività in HealthOmics, tale attività viene registrata in un CloudTrail evento insieme ad altri eventi AWS di servizio nella cronologia degli eventi. Puoi visualizzare, cercare e scaricare eventi recenti in Account AWS. Per ulteriori informazioni, consulta Visualizzazione degli eventi con la cronologia degli CloudTrail eventi.

Per una registrazione continua degli eventi del tuo Account AWS, inclusi gli eventi di HealthOmics, crea un percorso. Un trail consente di CloudTrail inviare file di log a un bucket Amazon S3. Per impostazione predefinita, quando si crea un percorso nella console, questo sarà valido in tutte le Regioni AWS. Il trail registra gli eventi di tutte le regioni della AWS partizione e consegna i file di log al bucket Amazon S3 specificato. Inoltre, puoi configurare altri AWS servizi per analizzare ulteriormente e agire in base ai dati sugli eventi raccolti nei log. CloudTrail Per ulteriori informazioni, consulta gli argomenti seguenti:

- Panoramica della creazione di un percorso
- CloudTrail servizi e integrazioni supportati
- Configurazione delle notifiche Amazon SNS per CloudTrail
- Ricezione di file di CloudTrail registro da più regioni e ricezione di file di CloudTrail registro da più account

Tutte HealthOmics le azioni vengono registrate CloudTrail e documentate nell'AWS HealthOmics API Reference. Ad esempio, le chiamate a `StartVariantImportJob` e `CreateReferenceStore` le `CreateWorkflow` azioni generano voci nei file di CloudTrail registro.

Ogni evento o voce di 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 utente IAM.
- 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 AWS servizio.

Per ulteriori informazioni, consulta Elemento CloudTrail userIdentity.

Comprensione delle HealthOmics voci dei file di registro

Un trail è una configurazione che consente la distribuzione di eventi come file di log in un bucket Amazon S3 specificato dall'utente. CloudTrail i file di registro contengono una o più voci di registro. Un evento rappresenta una singola richiesta proveniente da qualsiasi fonte e include informazioni sull'azione richiesta, la data e l'ora dell'azione, i parametri della richiesta e così via. CloudTrail i file di registro non sono una traccia ordinata dello stack delle chiamate API pubbliche, quindi non vengono visualizzati in un ordine specifico.

L'esempio seguente mostra una voce di CloudTrail registro che illustra l' CreateWorkflow azione.


```
{
    "eventVersion": "1.08",
    "userIdentity": {
        "type": "AssumedRole",
        "principalId": "AROAIU53LOGOMTOPXXNPG:username",
        "arn": "arn:aws:sts::account:assumed-role/admin/username",
        "accountId": "account-id",
        "accessKeyId": "accessKeyId",
        "sessionContext": {
            "sessionIssuer": {
                "type": "Role",
                "principalId": "AROAIU53LOGOMTOPXXNPG",
                "arn": "arn:aws:iam::account:role/admin",
                "accountId": "account",
                "userName": "admin"
            },
            "webIdFederationData": {},
            "attributes": {
                "creationDate": "2022-07-23T18:26:09Z",
                "mfaAuthenticated": "false"
            }
        }
    },
    "eventTime": "2022-07-23T18:46:42Z",
    "eventSource": "omics.amazonaws.com",
    "eventName": "CreateWorkflow",
    "awsRegion": "us-west-2",
    "sourceIPAddress": "205.251.233.176",
    "userAgent": "aws-cli/1.22.45 Python/3.9.13 Darwin/20.6.0 botocore/1.23.45",
    "requestParameters": {
        "name": "parameter_name",
```

```
        "definitionZip": "czM6Ly93b3JrZmxvd2RlZi1oZWxsby9kZWZpbml0aW9uLnppcA==",
        "requestId": "d788a73c-b81b-45fb-a8a6-d8bb4449ec8a"
    },
    "responseElements": {
        "id": "1002571",
        "arn": "arn:aws:omics:us-west-2:555555555555:instance/i-b188560f ",
        "status": "CREATING",
        "tags": {
            "resourceArn": "arn:aws:omics:us-west-2:083685709690:workflow/1002571"
        }
    },
    "requestID": "842d731d-f264-4b08-a2c9-2f7d45e1eaa3",
    "eventID": "76872ca2-f208-4193-807d-7dd7ea34e6b2",
    "readOnly": false,
    "eventType": "AwsApiCall",
    "managementEvent": true,
    "recipientAccountId": "083685709690",
    "eventCategory": "Management"
}
```


Utilizzo EventBridge con AWS HealthOmics

HealthOmics invia eventi ad Amazon EventBridge quando le risorse cambiano di stato. Le risorse includono lavori di importazione, processi di esportazione, condivisioni di risorse, flussi di lavoro, attività ed esecuzioni. Per ogni tipo di risorsa, esiste un elenco di modifiche di stato che generano un evento.

Un bus di eventi è un router che riceve eventi e li consegna alle destinazioni. Il tuo account include un bus di eventi predefinito che riceve automaticamente gli eventi dai AWS servizi. È possibile creare bus di eventi personalizzati aggiuntivi.

È possibile creare EventBridge regole per specificare le azioni da intraprendere quando il bus degli eventi riceve eventi. Ad esempio, è possibile creare una regola che notifichi le modifiche allo stato di una risorsa.

Gli scenari più comuni per l'utilizzo degli eventi includono:

- Per monitorare quando un utente condivide una risorsa con te o revoca la condivisione.
- Per controllare se un'esecuzione fallisce o viene completata correttamente.

Per ulteriori informazioni sull'utilizzo EventBridge, consulta What is Amazon EventBridge?

Argomenti

Configurato EventBridge per HealthOmics

Prima di poter monitorare gli EventBridge eventi, crea un EventBridge bus e crea regole per gli eventi di interesse.

Configura un EventBridge bus

È possibile utilizzare il bus eventi predefinito per il proprio bus eventi Account AWS o configurarne uno personalizzato. Per configurare un bus di eventi personalizzato, segui questi passaggi:

1. Apri la EventBridge console: https://console.aws.amazon.com/events/.
2. Nella barra di navigazione a sinistra, scegli Event bus.
3. Scegliere Create event bus (Crea bus di eventi).
4. Nel modulo Crea bus per eventi, inserisci un nome per il bus.
5. Scegliete Crea per creare il bus.

Crea una EventBridge regola

La procedura seguente mostra come creare una regola semplice. Per ulteriori informazioni sulle regole, vedere Regole in EventBridge.

1. Apri la EventBridge console: https://console.aws.amazon.com/events/.
2. Nel riquadro di navigazione di sinistra seleziona Rules (Regole).
3. Scegli Crea regola. La console apre il modulo Crea regola.
4. In Definisci i dettagli della regola, fornisci un nome per la regola.
    - In Nome, inserisci un nome per il bus.
    - Per Event bus, seleziona il bus per questa regola.
    - Scegli Next (Successivo).

5. In Build event pattern, in Event source seleziona Eventi AWS o eventi EventBridge partner.
6. Scorri verso il basso fino a Event pattern.
   a. Per Event source, seleziona i servizi AWS.
   b. Per il servizio AWS, inserisci omics nel filtro di testo e seleziona AWS HealthOmicscome servizio.
   c. Per Tipo di evento seleziona l'evento di interesse (o Tutti gli eventi).
   d. Scegli Next (Successivo).
7. In Seleziona obiettivi, seleziona un obiettivo per l'evento. Ad esempio, scegli il servizio AWS, il gruppo di CloudWatch log scelto e configura un gruppo di log.

   Per molti tipi di target, EventBridge necessita dell'autorizzazione per l'invio degli eventi. La console crea queste autorizzazioni per te.
8. (Facoltativo) In Configura tag, associa i tag alla regola.
9. In Rivedi e aggiorna, esamina la configurazione e scegli Crea regola.

EventBridge eventi in HealthOmics

La tabella seguente elenca gli eventi a EventBridge cui HealthOmics viene inviato e l'elenco dei possibili valori di stato per l'evento.

Nome evento	Valori di stato possibili
Modifica dello stato del processo di importazione delle annotazioni	Inviato, in corso, annullato, completato, non riuscito o completato con errori
Modifica dello stato di Annotation Store Share	In sospeso, in corso di attivazione, attivo, in eliminazione, eliminato, non riuscito
Modifica dello stato di Annotation Store	Creazione, creazione, aggiornamento, aggiornamento, eliminazione, eliminazione o creazione non riuscita
Modifica dello stato del processo di lettura Set Activation	Inviato, in corso, completato, non riuscito o completato con errori

Nome evento	Valori di stato possibili
Leggi la modifica dello stato di Set Export Job	Inviato, in corso, completato, non riuscito o completato con errori
Leggi la modifica dello stato del processo di importazione	Inviato, in corso, completato, non riuscito o completato con errori
Leggi Imposta modifica dello stato	Elaborazione del caricamento, caricamento non riuscito, attivo, archiviato, in corso di attivazione o eliminato
Modifica dello stato di Reference Import Job	Inviato, in corso, completato, non riuscito o completato con errori
Modifica dello stato di riferimento	Attivo o eliminato
Modifica dello stato del Reference Store	Creato, aggiornato, attivo o eliminato
Esegui la modifica dello stato	In sospeso, avviato, in esecuzione, interrotto, completato, eliminato, non riuscito o annullato
Modifica dello stato di Sequence Store	Creato, aggiornato, attivo o eliminato
Modifica dello stato dell'attività	In sospeso, avviato, in esecuzione, interrotto, completato, eliminato, non riuscito o annullato
Modifica dello stato del processo di importazi one delle varianti	Inviato, in corso, annullato, completato, non riuscito o completato con errori
Modifica dello stato di Variant Store Share	In sospeso, in corso di attivazione, attivo, in eliminazione, eliminato, non riuscito
Modifica dello stato del Variant Store	Creazione, creazione, aggiornamento, aggiornamento, eliminazione, eliminazione o creazione non riuscita
Modifica dello stato di condivisione del flusso di lavoro	In sospeso, in fase di attivazione, attivo, in eliminazione, eliminato, non riuscito

Nome evento	Valori di stato possibili
Modifica dello stato del flusso di lavoro	Creazione riuscita, creazione non riuscita, eliminazione riuscita o eliminazione non riuscita

Struttura del messaggio di evento

HealthOmics fornisce il massimo impegno nella consegna a cui inviare messaggi relativi agli eventi di modifica dello stato EventBridge. L'evento è un oggetto con struttura JSON che contiene anche dettagli sui metadati. È possibile utilizzare i metadati come input per ricreare l'evento o per ottenere ulteriori informazioni. Gli eventi includono i seguenti campi:

- `version`— Attualmente 0 (zero) per tutti gli eventi.
- `id`— Un UUID della versione 4 generato per ogni evento.
- `detail-type`— Il tipo di evento che viene inviato.
- `account`— L' Account AWS ID a 12 cifre del proprietario del bucket.
- `source`— Identifica il servizio che ha generato l'evento.
- `time`— L'ora in cui si è verificato l'evento.
- `region`— Identifica la parte Regione AWS del bucket.
- `resources`— Un array JSON che contiene l'Amazon Resource Name (ARN) del bucket.
- `detail`— Un oggetto JSON che contiene informazioni sull'evento.

Gli eventi Run includono i seguenti campi:

- `uuid`— L'identificatore universalmente univoco per la corsa.
- `workflowId`— Identificatore del flusso di lavoro associato a questa esecuzione.
- `workflowName`— Nome del flusso di lavoro associato a questa esecuzione.
- `runId`— Identificatore di esecuzione.
- `runName`— Nome della corsa.
- `runOutputUri`— L'URI in cui la corsa scriverà i dati di output.

Esempi di messaggi di evento

L'esempio seguente è un evento per una modifica dello stato di esecuzione, che mostra i campi aggiuntivi.


```
{
    "version":"0",
    "id":"c0e540f4-df38-b986-86c1-3e3730f971fe",
    "detail-type":"Run Status Change",
    "source":"aws.omics",
    "account":"123456789012",
    "time":"2022-10-20T22:07:35Z",
    "region":"us-west-2",
    "resources":[
        "arn:aws:omics:us-west-2:123456789012:run/2101313"
    ],
    "detail":{
        "omicsVersion":"1.0.0",
        "arn":"arn:aws:omics:us-west-2:123456789012:run/2101313",
        "status":"COMPLETED",
        "uuid":"153893cd-097a-40ec-aec7-838a97cd2b21",
        "runId": "1234567",
        "runName": "run name",
        "runOutputUri": "s3://amzn-s3-demo-bucket/run-output/2101313",
        "workflowId": "1234567",
        "workflowName": "workflow name"
    }
}
```


L'esempio seguente è un evento relativo a una modifica dello stato dell'attività.


```
{
    "version": "0",
    "id": "718d6817-c868-26d3-8ef0-0dc9b2ac73f4",
    "detail-type": "Task Status Change",
    "source": "aws.omics",
    "account": "123456789012",
    "time": "2024-10-30T09:05:44Z",
    "region": "us-west-2",
    "resources": ["arn:aws:omics:us-west-2:123456789012:task/8888888"],
    "detail": {
        "omicsVersion": "1.0.0",
        "arn": "arn:aws:omics:us-west-2:123456789012:task/8888888",
```

```
        "status": "COMPLETED",
        "runArn": "arn:aws:omics:us-west-2:123456789012:run/2101313",
        "runUuid": "153893cd-097a-40ec-aec7-838a97cd2b21",
        "runId": "1234567",
        "runName": "run name",
        "workflowId": "1234567",
        "workflowName": "workflow name"
    }
}
```

Di seguito è riportato un esempio di evento relativo alla modifica dello stato di un set di lettura.

```
{
  "version": "0",
  "id": "64ca0eda-9751-dc55-c41a-1bd50b4fc9b7",
  "detail-type": "Read Set Status Change",
  "source": "aws.omics",
  "account": "123456789012",
  "time": "2023-04-04T17:53:06Z",
  "region": "us-west-2",
  "resources": ["arn:aws:omics:us-west-2:123456789012:sequenceStore/1234567890/
readSet/3456789012"],
  "detail": {
     "omicsVersion": "1.0.0",
     "arn": "arn:aws:omics:us-west-2:123456789012:sequenceStore/1234567890/
readSet/3456789012",
     "sequenceStoreId" : "1234567890",
     "id": "3456789012",
     "status": "PROCESSING_UPLOAD"
  }
}
```

Un evento simile viene creato per un processo di importazione di un negozio di varianti.

```
{
    "version": "0",
    "id": "6a7e8feb-b491-4cf7-a9f1-bf3703467718",
    "detail-type": "Variant Store Status Change",
    "source": "aws.omics",
    "account": "123456789012",
    "time": "2015-12-22T18:43:48Z",
    "region": "us-east-1",
    "resources": ["arn:aws:omics:us-east-1:123456789012:myvariantstore2"],
```

```
    "detail": {
       "omicsVersion": "1.0.0",
       "arn": "arn:aws:omics:us-east-1:123456789012:myvariantstore2",
       "status": "CREATED",
       "storeId": "6710c5f02610",
       "storeName": "myvariantstore2"
    }
}
```

Di seguito è riportato un evento relativo a una modifica dello stato del processo di importazione.

```
{
    "version": "0",
    "id": "6a7e8feb-b491-4cf7-a9f1-bf3703467718",
    "detail-type": "Variant Import Job Status Change",
    "source": "aws.omics",
    "account": "123456789012",
    "time": "2015-12-22T18:43:48Z",
    "region": "us-east-1",
    "resources": ["arn:aws:omics:us-east-1:123456789012:my_variant_store/
b64ea9a3-459f-4b68-92c3-3ddb83209fe9"],
    "detail": {
       "omicsVersion": "1.0.0",
       "arn": "arn:aws:omics:us-east-1:123456789012:my_variant_store/
b64ea9a3-459f-4b68-92c3-3ddb83209fe9",
       "status": "COMPLETED",
       "jobId": "b64ea9a3-459f-4b68-92c3-3ddb83209fe9",
       "storeId": "a74869f91e20",
       "storeName": "my_variant_store"
    }
}
```

Risoluzione dei problemi

I seguenti argomenti possono aiutarti a risolvere i problemi che si verificano durante l'utilizzo di flussi di lavoro e archivi di dati. HealthOmics

Argomenti

Risoluzione dei flussi di lavoro

Argomenti

Come posso risolvere un'esecuzione non riuscita?

Utilizza l'operazione GetRunAPI per recuperare il motivo dell'errore. Per ulteriori informazioni, consulta Motivi dell'errore di esecuzione.

Come posso risolvere un'operazione non riuscita?

Esamina il codice di errore contenuto nel messaggio di errore dell'attività per comprendere l'errore. Controlla i registri dell'attività CloudWatch per visualizzare i messaggi di registrazione dettagliati relativi all'attività. Se non ricevi messaggi di registro dettagliati, puoi modificare il flusso di lavoro per generare istruzioni di registro aggiuntive. Per ulteriori informazioni, consulta Monitoraggio HealthOmics con CloudWatch registri.

Dove posso trovare i registri del motore relativi alle esecuzioni completate con successo?

HealthOmics pubblica i log solo CloudWatch per le esecuzioni non riuscite. Se un'esecuzione viene completata correttamente, HealthOmics invia i log del motore al tuo bucket Amazon S3. Per ulteriori informazioni, consulta Accedi ad Amazon S3.

Come posso ridurre la dimensione dei parametri di input per un flusso di lavoro?

È possibile specificare fino a 50 KB di parametri di input per un flusso di lavoro. È possibile utilizzare le importazioni di directory o i fogli di esempio per rispettare questo limite di dimensione. Per ulteriori informazioni, consulta Gestione delle dimensioni dei parametri di esecuzione.

Perché la mia corsa non si completa?

Se ci sono problemi con il codice e i processi non sono terminati correttamente, l'esecuzione potrebbe non rispondere o «bloccarsi». Per ulteriori informazioni su come prevenire e catturare le corse che non rispondono, vedere. Guida per le esecuzioni che non rispondono

Risoluzione dei problemi relativi alla memorizzazione nella cache delle chiamate

I seguenti argomenti possono aiutarti a risolvere i problemi che si verificano con la memorizzazione nella cache delle chiamate.

Argomenti

Perché la mia corsa non viene salvata nella cache?

1. Verifica che l'esecuzione sia configurata per utilizzare una cache controllando il campo CacheID nella risposta dell'operazione GetRun API. Utilizzando la CLI, esegui questo comando:. `aws omics get-run –id <run_id>`

2. Se l'esecuzione è andata a buon fine, verifica che il comportamento della cache restituito nella GetRun risposta sia CACHE_ALWAYS. Se il comportamento della cache è impostato su CACHE_ON_FAILURE, le esecuzioni verranno salvate nella cache solo quando falliscono.

Perché un'attività non utilizza la voce della cache?

<cache_id><cache_uuid>Nel gruppo di `/aws/omics/WorkflowLog` CloudWatch log, apri il flusso di log per la cache di esecuzione: runCache//.

1. Verificate che un'esecuzione precedente abbia creato una voce nella cache per l'operazione che vi aspettavate fosse memorizzata nella cache. Le esecuzioni salvate nella cache verranno registrate con un messaggio di registro CACHE_ENTRY_CREATED.
2. Individuate il registro CACHE_MISS relativo all'attività ed eseguitela completata. Se non è presente alcuna voce di registro, verificate che l'esecuzione sia stata configurata per utilizzare la cache.
3. Se è stata creata una voce nella cache, verificate che la CPUs memoria GPUs e il container digest siano identici per entrambe le attività. L'ARN dell'attività che ha creato la voce della cache si trova nel messaggio di registro.
4. Se i requisiti di calcolo per entrambe le attività corrispondono, verifica che gli input non siano cambiati tra le attività. Per fare ciò, apri i registri del motore. Se l'esecuzione ha lo stato FAILED, i log si troveranno in Cloudwatch Log Group/. aws/omics/WorkflowLog Altrimenti, i log del motore possono essere trovati nella directory di output dell'esecuzione.

Perché la memorizzazione nella cache delle chiamate per un'attività è disabilitata?

Verifica se l'attività è configurata per disattivare la memorizzazione nella cache utilizzando le funzionalità del motore di workflow:

- Per i flussi di lavoro WDL: controlla se l'opzione Volatile è impostata su nella sezione meta `true`
- Per i flussi di lavoro Nextflow: controlla se l'attività ha la direttiva cache impostata su `false`
- Per i flussi di lavoro CWL: controlla se l'attività ha impostato EnableReuse per la funzionalità `false` WorkReuse

Risoluzione dei problemi degli archivi dati

Argomenti

Perché S3 GetObject non funziona sul mio set di lettura?

Nella maggior parte dei casi, l'errore è dovuto alla mancanza di un'autorizzazione. L'autorizzazione di lettura di Sequence Store S3 è una configurazione bidirezionale che richiede sia la policy di accesso di Sequence Store S3 per consentire l'accesso sia la policy di accesso associata al principale IAM una policy che consenta l'accesso. Per maggiori dettagli sui requisiti della policy, consulta. Autorizzazioni per l'accesso ai dati tramite Amazon S3 URIs Verificate che siano presenti le seguenti configurazioni:

- La policy di accesso di Sequence Store S3 ha consentito esplicitamente l'accesso al principale IAM o alla radice dell'account del principale.
- Verifica che il principale IAM disponga di una politica che fornisca esplicitamente l'autorizzazione alla risorsa a cui si accede. Tieni presente che la policy principale di IAM deve utilizzare l'ARN del punto di accesso e non il percorso basato sull'alias del punto di accesso quando definisce le autorizzazioni e che l'ARN è nella condizione e non viene utilizzato per specificare una risorsa.
- Se il tuo negozio utilizza una chiave gestita dal cliente (CMK-KMS), assicurati che il responsabile IAM disponga delle autorizzazioni di decrittografia sulla chiave. kms: Consulta la guida all'accesso KMS su più account per configurare l'utilizzo tra account.

Se hai una politica che utilizza controlli di accesso basati su tag, assicurati quanto segue:

- Assicurati che l'archivio delle sequenze abbia terminato la sincronizzazione dei tag. Per questo, lo stato del negozio deve essere active e non updating esserlo.
- Assicurati che non vi siano errori di battitura nella chiave del tag o nel valore della chiave sul set di lettura e sulla policy.

Perché non riesco a vedere il mio negozio di annotazioni o il mio negozio di varianti in Athena?

In Lake Formation, assicurati di creare un link alla risorsa basato sul negozio che è stato condiviso con te. Dopo aver creato un link a una risorsa a cui sei autorizzato ad accedere, il negozio dovrebbe essere visibile in Athena. Per ulteriori informazioni, consulta Configurazione di Lake Formation per l'uso HealthOmics.

Perché non riesco ad accedere al mio archivio dati in Athena?

Se il tuo archivio di annotazioni o varianti è visibile ma ricevi un messaggio di errore che indica che l'accesso è negato, controlla quale versione del motore di query stai utilizzando. Sono supportate solo le query eseguite utilizzando la versione 3 del motore. Per ulteriori informazioni sulle versioni del motore di query Athena, consulta la documentazione di Amazon Athena.

Risoluzione dei problemi con Amazon Q CLI

La CLI di Amazon Q può aiutarti a semplificare il processo di risoluzione dei problemi tramite:

- Analisi delle esecuzioni del flusso di lavoro e degli errori delle attività di debug
- Raccolta di registri e messaggi di errore pertinenti
- Creazione di casi di AWS supporto con tutti i registri di debug necessari allegati
- Elimina le informazioni di identificazione personale (PII) dalle informazioni inviate a Support AWS

Per ulteriori informazioni sull'utilizzo dell'interfaccia a riga di comando di Amazon Q AWS HealthOmics per la risoluzione dei problemi e la creazione di casi di supporto, consulta il tutorial sull'intelligenza artificiale generativa di HealthOmics Agentic su. GitHub

> **Warning**
>
> Quando lavori con Amazon Q CLI, esamina tutti i contenuti generati e le azioni proposte prima di procedere. Fornisci feedback per migliorare la qualità della risposta e soddisfare i requisiti del tuo flusso di lavoro. Per ulteriori informazioni, consulta Considerazioni sulla sicurezza e best practice per Amazon Q.

Quote per AWS HealthOmics

AWS compila il tuo account con i valori predefiniti per le HealthOmics quote. Salvo diversa indicazione, ogni valore di quota è il valore massimo per regione.

> **Important**
>
> È possibile richiedere aumenti alla maggior parte delle quote di servizio e delle quote API. Per ulteriori informazioni, consulta gli argomenti seguenti:

Argomenti

HealthOmics quote di servizio

La tabella seguente elenca le quote HealthOmics di servizio, insieme ai relativi valori predefiniti. Per visualizzare le quote correnti per ogni regione, apri la console Service Quotas.

> **Important**
>
> Puoi richiedere un aumento fino a una quota regolabile utilizzando la console Service Quotas.

Per ulteriori informazioni sulle quote di servizio, vedere Richiesta di un aumento della quota nella Service Quotas User Guide. Per una quota che non è disponibile nella console Service Quotas, utilizza il modulo di aumento della quota.

Name	Predefinita	Adattabile	Description
Analytics: numero massimo di archivi di annotazioni	Ogni regione supportata: 10	Sì	Il numero massimo di archivi di annotazioni

Name	Predefinita	Adatta e	Description
			nella regione corrente AWS
Analytics: numero massimo di lavori di importazione simultanei di varianti o archivi di annotazioni	Ogni regione supportata: 5	Sì	Il numero massimo di processi di importazione simultanei nella regione corrente AWS
Analytics: numero massimo di file per processo di importazione dell'archivio varianti	Ogni regione supportata: 1.000	Sì	Il numero massimo di file per processo di importazione di varianti nella AWS regione corrente
Analytics: numero massimo di condivisioni per archivio di annotazioni	Ogni regione supportata: 10	Sì	Il numero massimo di condivisioni per archivio di annotazioni nella regione corrente AWS
Analytics: numero massimo di condivisioni per archivio di varianti	Ogni regione supportata: 10	Sì	Il numero massimo di condivisioni per archivio di varianti nella AWS regione corrente
Analytics: dimensione massima di ogni file in un processo di importazione di varianti	Ogni regione supportata: 20 GB	Sì	La dimensione massima di un file in un processo di importazione di varianti nella AWS regione corrente
Analytics: dimensione massima di ogni file in un processo di importazione di annotazioni	Ogni regione supportata: 20 GB	Sì	La dimensione massima di un file in un processo di importazione di annotazioni nella regione corrente AWS

Name	Predefinita	Adatta e	Description
Analytics: numero massimo di archivi di varianti	Ogni regione supportata: 10	Sì	Il numero massimo di archivi di varianti nella AWS regione corrente
Analytics: numero massimo di versioni per archivio di annotazioni	Ogni regione supportata: 10	Sì	Il numero massimo di versioni per archivio di annotazioni nella regione corrente AWS
Configurazioni: numero massimo di configurazioni	Ogni regione supportata: 10	Sì	Il numero massimo di configurazioni nella regione corrente. AWS
Archiviazione: numero massimo di processi di attivazione simultanei di set di lettura	Ogni regione supportata: 25	Sì	Il numero massimo di processi di attivazione simultanei di set di lettura nella regione corrente AWS
Archiviazione: numero massimo di lavori di esportazione simultanei in sequenza e nell'archivio di riferimento	Ogni regione supportata: 5	Sì	Il numero massimo di processi di esportazione simultanei da una sequenza o da un archivio di riferimenti nella regione corrente AWS
Archiviazione: numero massimo di lavori di importazione simultanei di sequenze o archivi di riferimento	Ogni regione supportata: 5	Sì	Il numero massimo di processi di importazione simultanei per una sequenza o un archivio di riferimenti nella regione corrente AWS

Name	Predefinita	Adatta e	Description
Archiviazione: numero massimo di set di lettura per archivio di sequenze	Ogni regione supportata: 1.000.000	Sì	Il numero massimo di set di lettura in un archivio di sequenza nella AWS regione corrente
Archiviazione: numero massimo di riferimenti per archivio di riferimento	Ogni Regione supportata: 50	Sì	Il numero massimo di riferimenti in un archivio di riferimenti nella AWS regione corrente
Archiviazione: numero massimo di archivi in sequenza	Ogni regione supportata: 20	Sì	Il numero massimo di sequenze memorizza te nella AWS regione corrente
Flussi di lavoro: numero massimo di attività GPUs	Ogni regione supportata: 12	Sì	Il numero massimo di attivi simultanei GPUs nella regione corrente AWS . In us-east-1 e us-west-2, le richieste di aumento delle quote per valori fino a 500 vengono approvate automatic amente.

Name	Predefinita	Adattabile	Description
Flussi di lavoro: numero massimo di esecuzioni attive simultanee utilizzando Dynamic Run Storage	Ogni Regione supportata: 50	Sì	Il numero massimo di esecuzioni attive che utilizzano l'archiviazione di esecuzione dinamica nella regione corrente AWS . Le richieste di aumento delle quote per valori fino a 200 vengono approvate automatic amente.
Flussi di lavoro: numero massimo di esecuzioni attive simultanee utilizzando lo storage a esecuzione statica	Ogni regione supportata: 10	Sì	Il numero massimo di esecuzioni attive che utilizzano l'archiviazione di esecuzione statica nella regione corrente AWS . Le richieste di aumento delle quote per valori fino a 50 vengono approvate automatic amente.
Flussi di lavoro: numero massimo di attività simultanee per esecuzione	Ogni regione supportata: 25	Sì	Il numero massimo di attività simultanee in ciascuna esecuzione nella regione corrente. AWS In us-east-1 e us-west-2, le richieste di aumento delle quote per valori fino a 100 vengono approvate automatic amente.

Name	Predefinita	Adatta e	Description
Flussi di lavoro: durata massima di esecuzione	Ogni regione supportata: 604.800 secondi	Sì	La durata massima dell'esecuzione del flusso di lavoro nella regione corrente. AWS
Flussi di lavoro: numero massimo di esecuzioni (attive o inattive)	Ogni regione supportata: 100.000	Sì	Il numero massimo di esecuzioni (attive o inattive) nella regione corrente. AWS
Flussi di lavoro: numero massimo di condivisioni per flusso di lavoro	Ogni regione supportata: 100	Sì	Il numero massimo di condivisioni per flusso di lavoro nella regione corrente AWS
Flussi di lavoro: capacità di storage statica massima per esecuzione	Ogni regione supportata: 9.600	Sì	La capacità di archiviaz ione statica massima in gibibyte (GiB) per ogni esecuzione nella regione corrente. AWS In us-east-1 e us-west-2, le richieste di aumento delle quote per valori fino a 50.000 vengono approvate automatic amente.
Flussi di lavoro: flussi di lavoro massimi	Ogni regione supportata: 1.000	Sì	Il numero massimo di flussi di lavoro nella regione corrente. AWS

Name	Predefinita	Adatta e	Description
Flussi di lavoro: transazioni al secondo (TPS) per l'operazione StartRun	Ogni regione supportata: 1	Sì	Il numero massimo di transazioni al secondo (TPS) per l' StartRun operazione nella regione corrente. AWS

HealthOmics quote a dimensione fissa

Oltre aHealthOmics quote di servizio, HealthOmics include quote con dimensioni fisse. Non è possibile richiedere un aumento per questi valori.

Salvo diversa indicazione, ogni quota elenca il valore massimo per regione.

Argomenti

- HealthOmics analisi: quote a dimensione fissa.
- HealthOmics quote di archiviazione a dimensione fissa
- HealthOmics quote a dimensione fissa per il flusso di lavoro
- HealthOmics Quote a dimensione fissa del flusso di lavoro Ready2Run

HealthOmics analisi: quote a dimensione fissa.

La tabella seguente mostra i valori massimi supportati per le quote di analisi. Questi valori non sono regolabili.

Nome	Description	Massimo	Regolabile Sì/No
Analytics: numero massimo di file per processo di importazione dell'archivio di annotazioni	Il numero massimo di file per processo di importazione delle annotazioni.	1	No

HealthOmics quote di archiviazione a dimensione fissa

La tabella seguente mostra i valori massimi supportati per i file di archiviazione. Questi valori non sono regolabili.

Nome	Description	Massimo	Regolabile Sì/No
Storage: dimensione massima della policy di accesso alle risorse di accesso S3	Dimensione massima della politica delle risorse di accesso S3	15 KB	No
Archiviazione: numero massimo di tag a livello di set propagati	Il numero massimo di chiavi di tag a livello di set, per archivio, che si propagano all'oggetto S3	5	No
Archiviazione: numero massimo di set di lettura per processo di attivazione	Il numero massimo di set di lettura per processo di attivazione.	20	No
Archiviazione: numero massimo di set di lettura per processo di esportazione	Il numero massimo di set di lettura per processo di esportazione.	100	No
Archiviazione: numero massimo di set di lettura per processo di importazione	Il numero massimo di set di lettura per processo di importazione.	100	No
Archiviazione: numero massimo di archivi di riferimento	Il numero massimo di negozi di riferimento.	1	No
Archiviazione: dimensione massima	La dimensione massima della parte	100 MB	No

Nome	Description	Massimo	Regolabile Sì/No
della parte per un caricamento diretto	per il caricamento diretto in un archivio di sequenze.		
Archiviazione: numero massimo di parti nel file per il caricamento diretto	Il numero massimo di parti in un file per il caricamento diretto in un archivio di sequenze.	10.000	No
Archiviazione: dimensione massima di riferimento	La dimensione massima di un file di riferimento che può essere importato in un archivio di riferimento.	15 GB	No
Archiviazione: dimensione massima della sorgente impostata per la lettura	La dimensione massima di un singolo file sorgente in un set di lettura che può essere importato in un archivio di sequenze.	976 GB	No

HealthOmics quote a dimensione fissa per il flusso di lavoro

La tabella seguente mostra i valori massimi supportati per le quote del flusso di lavoro. Questi valori non sono regolabili.

Nome	Description	Dimensione massima	Regolabile Sì/No
Flussi di lavoro: numero massimo di gruppi di esecuzione	Il numero massimo di gruppi di esecuzione.	1000	No

Nome	Description	Dimensione massima	Regolabile Sì/No
Flussi di lavoro: numero massimo di cache di esecuzione	Il numero massimo di cache di esecuzione che è possibile creare per un account. Una o più esecuzioni possono condividere la stessa cache di esecuzione. Non esiste una quota per il numero di esecuzioni che HealthOmics possono essere memorizzate nella cache per account.	1000	No
Flussi di lavoro: numero massimo di versioni del flusso di lavoro	Il numero massimo di versioni del flusso di lavoro per flusso di lavoro.	1000	No
Flussi di lavoro: dimensione del contenitore dell'istanza della CPU	La dimensione massima dell'immagine del contenitore per un'istanza di CPU.	45 GiB	No
Flussi di lavoro: dimensione del contenitore dell'istanza GPU	La dimensione massima dell'immagine del contenitore per un'istanza GPU.	95 GiB	No
Memoria condivisa dell'istanza GPU /dev/shm	La quantità massima di memoria condivisa per istanza GPU.	8 GB per GPU	No

Nome	Description	Dimensione massima	Regolabile Sì/No
Flussi di lavoro: esegui il file dei parametri	La dimensione massima di un file dei parametri di esecuzione.	50.000 byte	No
Flussi di lavoro: file modello dei parametri del flusso di lavoro	Il numero massimo di voci e la dimension e massima del file per un file modello di parametri di workflow. Questa quota si applica ai flussi di lavoro creati utilizzan do la console o l'API.	1.000 voci, 400 KB	No
Flussi di lavoro - Dimensione del file di definizione del flusso di lavoro - API	La dimensione massima del file di definizione del flusso di lavoro quando si crea il flusso di lavoro utilizzando l'operazi one API o un AWS SDK.	100 MB	No
Flussi di lavoro - Dimensione del file di definizione del flusso di lavoro - Console (caricamento diretto)	La dimensione massima del file di definizione del flusso di lavoro che puoi fornire come caricamento diretto, quando crei il flusso di lavoro utilizzando la console.	4,4 MB	No

Nome	Description	Dimensione massima	Regolabile Sì/No
Flussi di lavoro - Dimensione del file di definizione del flusso di lavoro - Console (caricamento da Amazon S3)	La dimensione massima del file di definizione del flusso di lavoro che puoi fornire come caricamento da Amazon S3, quando crei il flusso di lavoro utilizzando la console.	100 MB	No
Flussi di lavoro: dimensioni del repository	La dimensione massima di un archivio di codice esterno.	1 GiB	No
Flussi di lavoro: dimensione dei singoli file del repository	La dimensione massima di un singolo file proveniente da un archivio di codice esterno.	100 MiB	No
Flussi di lavoro: dimensione del file README	La dimensione massima di un file README.	500 KiB	No

Per suggerimenti su come ridurre le dimensioni del file dei parametri di esecuzione, consultaGestione delle dimensioni dei parametri di esecuzione.

HealthOmics Quote a dimensione fissa del flusso di lavoro Ready2Run

Ogni flusso di lavoro Ready2Run ha una dimensione massima del file di input. Nella tabella seguente, le unità di dimensione dei file sono elencate in Gibibyte (GiB). Queste dimensioni massime dei file non sono regolabili.

Nome del flusso di lavoro Ready2Run	Dimensione massima del file di input (GiB)	Regolabile (Sì/No)
AlphaFold per 601-1200 residui	1	No
AlphaFold per un massimo di 600 residui	1	No
Bases2Fastq per 2x150	1000	No
Bases2Fastq per 2x300	1000	No
Bases2Fastq per 2x75	500	No
ESMFold per un massimo di 800 residui	1	No
GATK-BP fq2bam	64	No
GATK-BP Germline bam2vcf per genoma 30x	39	No
GATK-BP Germline fq2vcf per genoma 30x	64	No
GATK-BP WES somatico bam2vcf	86	No
NVIDIA Parabricks BAM2 FQ2 BAM WGS per un massimo di 30 volte	80	No
NVIDIA Parabricks BAM WGS per un massimo di 50 volte BAM2 FQ2	120	No

Nome del flusso di lavoro Ready2Run	Dimensione massima del file di input (GiB)	Regolabile (Sì/No)
NVIDIA Parabricks BAM WGS per un massimo di 5 volte BAM2 FQ2	20	No
NVIDIA Parabricks BAM WGS per un massimo di 30 volte FQ2	71	No
NVIDIA Parabricks BAM WGS per un massimo di 50X FQ2	137	No
NVIDIA Parabricks BAM WGS per un massimo di 5 volte FQ2	13	No
NVIDIA Parabricks Germline WGS per un massimo di 30 volte DeepVariant	71	No
NVIDIA Parabricks Germline WGS per un massimo di 50X DeepVariant	137	No
NVIDIA Parabricks Germline WGS per un massimo di 5 volte DeepVariant	12	No
NVIDIA Parabricks Germline WGS per un massimo di 30 volte HaplotypeCaller	71	No
NVIDIA Parabricks Germline WGS per un massimo di 50X HaplotypeCaller	137	No

Nome del flusso di lavoro Ready2Run	Dimensione massima del file di input (GiB)	Regolabile (Sì/No)
NVIDIA Parabricks Germline WGS per un massimo di 5 volte HaplotypeCaller	13	No
NVIDIA Parabricks Somatic Mutect2 WGS per un massimo di 50 volte	196	No
sc RNAseq con Kallisto BUStools	119	No
sc RNAseq con salmone Alevin-fry	119	No
sc RNAseq con STARsolo	119	No
Sentieon Germline BAM WES per un massimo di 300 volte	9	No
Sentieon Germline BAM WGS per un massimo di 32x	18	No
Sentieon Germline FASTQ WES per un massimo di 100x	5	No
Sentieon Germline FASTQ WES per un massimo di 300 volte	26	No
Sentieon Germline FASTQ WGS per un massimo di 32x	51	No
LongRead Sentieon per PNG	25	No
Sentieon per LongRead PacBio HiFi	58	No

Nome del flusso di lavoro Ready2Run	Dimensione massima del file di input (GiB)	Regolabile (Sì/No)
Sentieon Somatic WES	50	No
Sentieon Somatic SGS	113	No
Ultima DeepVariant Genomics per un massimo di 40 volte	91	No

HealthOmics Quote API

HealthOmics ha le seguenti quote relative alle operazioni API. Dove indicato, la quota è regolabile. Per richiedere un aumento, utilizza il modulo di aumento della quota.

Per ogni operazione API elencata, la quota è il numero massimo di transazioni al secondo (TPS) per quell'operazione API in ciascuna regione.

Argomenti

- Quote API generali
- Quote API di archiviazione
- Quote API del flusso di lavoro
- Quote delle API di analisi

Quote API generali

La tabella seguente elenca le operazioni API generali che si applicano a più di una categoria (archiviazione, flussi di lavoro e analisi).

Operazione API	TPS massimo predefinito	Regolabile (Sì/No)
AcceptShare, CreateShare, DeleteShare, GetShare, ListShares	1 TPS	Sì

Quote API di archiviazione

La tabella seguente elenca le operazioni dell'API di archiviazione.

Funzionamento dell'API di archiviazione	TPS massimo predefinito	Regolabile (Sì/No)
CreateSequenceStore, UpdateSequenceStore, DeleteSequenceStore, CreateReferenceStore, DeleteReferenceStore	1 TPS	Sì
BatchDeleteReadSet, DeleteReference	1 TPS	Sì
CreateMultipartReadSetUpload, CompleteMultipartReadSetUpload, AbortMultipartReadSetUpload	1 TPS	No
Ottiene S3, inserisce S3, Elimina S3 AccessPolicy AccessPolicy AccessPolicy	1 TPS	Sì
GetReference	10 TPS	Sì
UploadReadSetPart	10 TPS	Sì
GetReadSet	30 TPS	Sì
GetSequenceStore, ListSequenceStores	5 TPS	Sì
GetReadSetMetadata, ListReadSets	5 TPS	Sì

Funzionamento dell'API di archiviazione	TPS massimo predefinito	Regolabile (Sì/No)
StartReadSetImportJob, GetReadSetImportJob, ListReadSetImportJobs	5 TPS	Sì
StartReadSetExportJob, GetReadSetExportJob, ListReadSetExportJobs	5 TPS	Sì
ListReferenceStores	5 TPS	Sì
StartReferencetImportJob, GetReferenceImportJob, ListReferenceImportJobs	5 TPS	Sì
ListReferences, GetReferenceMetadata	5 TPS	Sì
StartReadsetActivationJob	5 TPS	Sì
ListReadsetActivationJobs, GetReadSetActivationJob	5 TPS	Sì
ListMultipartReadSetUploads, ListReadSetUploadParts	5 TPS	Sì
TagResource, UntagResource, ListTagsForResource	5 TPS	Sì

Quote API del flusso di lavoro

La tabella seguente elenca le operazioni dell'API del flusso di lavoro.

Funzionamento dell'API Workflow	TPS massimo predefinito	Regolabile (Sì/No)
StartRun	1 TPS	Sì
CreateWorkflow	5 TPS	Sì
CancelRun, DeleteRun , GetRun, GetRunTask, ListRunTasks, ListRuns	10 TPS	Sì
CreateRunGroup, DeleteRun Group, GetRunGroup, ListRunGroups, UpdateRun Group	10 TPS	Sì
CreateRunCache, UpdateRun Cache, DeleteRunCache, GetRunCache, ListRunCaches	10 TPS	Sì
DeleteWorkflow, GetWorkfl ow, ListWorkflows, UpdateWor kflow	10 TPS	Sì

Quote delle API di analisi

La tabella seguente elenca le operazioni dell'API di analisi.

Funzionamento dell'API di analisi	TPS massimo predefinito	Regolabile (Sì/No)
CreateVariantStore, DeleteVar iantStore, GetVariantStore, ListVariantStores, UpdateVar iantStore	1 TPS	No
StartVariantImportJob, CancelVariantImportJob,	1 TPS	No

Funzionamento dell'API di analisi	TPS massimo predefinito	Regolabile (Sì/No)
GetVariantImportJob, ListVariantImportJobs		
CreateAnnotationStore, DeleteAnnotationStore, GetAnnotationStore, ListAnnotationStores, UpdateAnnotationStore	1 TPS	No
StartAnnotationImportJob, ListAnnotationImportJobs, GetAnnotationImportJob, CancelAnnotationImportJob	1 TPS	No

Cronologia dei documenti per la Guida per HealthOmics l'utente

La tabella seguente descrive le versioni della documentazione per HealthOmics.

Modifica	Descrizione	Data
AWS HealthOmics i negozi di varianti e gli archivi di annotazioni non sono più aperti a nuovi clienti.	AWS HealthOmics i negozi di varianti e i negozi di annotazio ni non sono più aperti a nuovi clienti. Per ulteriori informazioni, consulta Modifica della disponibilità di AWS HealthOmics Variant Store e Annotation Store.	7 novembre 2025
AWS HealthOmics i negozi di varianti e i negozi di annotazio ni non saranno più aperti ai nuovi clienti a partire dal 7 novembre 2025.	AWS HealthOmics i negozi di varianti e i negozi di annotazio ni non saranno più aperti a nuovi clienti a partire dal 7 novembre 2025. Se desideri utilizzare gli store di varianti o gli archivi di annotazioni, registrati prima di tale data. I clienti esistenti possono continuare a utilizzare il servizio normalmente. Per ulteriori informazioni, consulta Modifica della disponibilità del negozio di AWS HealthOmi cs varianti e dell'archivio di annotazioni.	7 ottobre 2025
Nuove funzionalità	HealthOmics ha aggiunto il supporto per i flussi di lavoro per sincronizzare un repositor	28 agosto 2025

	y Amazon ECR privato con un registro upstream. Per ulteriori informazioni, consulta Container images for private workflows in. HealthOmics	
Nuove funzionalità di integrazione tra README e repository	È stato aggiunto il supporto per la creazione di flussi di lavoro da archivi di codice esterni e file README.	24 luglio 2025
Nuove funzionalità	HealthOmics ha aggiunto il supporto per l'interpolazione automatica dei parametri di Nextflow. Per ulteriori informazioni, consulta File modello di parametri per i flussi di lavoro. HealthOmics	27 giugno 2025
Nuove funzionalità	HealthOmics ha aggiunto il supporto per i flussi di lavoro per interpolare i parametri di esecuzione da un file di definizione del flusso di lavoro WDL. Per ulteriori informazioni, consulta File modello di parametri per i flussi di lavoro. HealthOmics	30 maggio 2025
Nuove funzionalità	HealthOmics ha aggiunto il supporto per il controllo delle versioni del flusso di lavoro. Per ulteriori informazioni, consulta Controllo delle versioni del flusso di lavoro in. HealthOmics	18 aprile 2025

Nuove funzionalità	HealthOmics ha aggiunto un throughput elastico per lo storage dinamico. Per ulteriori informazioni, consulta Run storage types in HealthOmics.	16 aprile 2025
Nuove funzionalità	HealthOmics ha aggiunto controlli di accesso basati sugli attributi per le sedi Sequence Store S3 e la possibilità di sincronizzare fino a cinque tag di lettura con un oggetto Sequence Store S3. Per ulteriori informazioni, consulta Creazione di un archivio di sequenze. HealthOmics	22 novembre 2024
Nuove funzionalità	HealthOmics ha aggiunto il supporto per la memorizza zione nella cache delle chiamate, nota anche come curriculum, per i flussi di lavoro privati. Per ulteriori informazi oni, consulta Call caching.	20 novembre 2024
Nuove funzionalità	HealthOmics sono stati aggiunti nuovi campi API per aiutarti a mappare tra i processi di input dell'archivio sequenziale e i set di lettura.	29 agosto 2024
Nuove funzionalità	HealthOmics ha aggiunto il supporto per la gestione delle versioni di Nextflow. Per saperne di più, consulta le versioni di Nextflow.	14 agosto 2024

Nuove funzionalità	HealthOmics ha aggiunto il supporto per flussi di lavoro condivisi e storage dinamico.	30 aprile 2024
Nuove funzionalità	HealthOmics ha aggiunto il supporto per l'accesso di Amazon S3 agli archivi di riferimento e di sequenza e il supporto per. SHA256 ETags	15 aprile 2024
Nuove funzionalità	HealthOmics tag di entità aggiunti (ETags) per gli archivi di sequenze.	6 ottobre 2023
Nuove funzionalità	HealthOmics ha aggiunto il controllo delle versioni dell'archivio di annotazioni e la condivisione analitica degli archivi.	15 agosto 2023
Nuove funzionalità	HealthOmics ha aggiunto Common Workflow Language (CWL) come linguaggi o supportato per i flussi HealthOmics di lavoro.	30 giugno 2023
Nuove funzionalità	HealthOmics ha aggiunto nuovi flussi di lavoro Ready2Run, supporto GPU per i flussi di lavoro, analisi dei dati per gli archivi di annotazio ni, caricamento diretto nello storage e integrazione con. HealthOmics EventBridge	15 maggio 2023

Nuova politica gestita	HealthOmics ha aggiunto una nuova politica gestita che fornisce l'accesso completo. Per ulteriori informazioni, consulta le policy gestite da AWS.	23 febbraio 2023
Nuova policy gestita	HealthOmics ha aggiunto una nuova politica gestita che limita l'accesso alla sola lettura. Per ulteriori informazioni, consulta le policy gestite da AWS.	29 novembre 2022
Versione iniziale	Versione iniziale della Guida per l' HealthOmics utente	29 novembre 2022

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.