

Nf-core ?????? ??????? ?? (spatialvi)

2025 5 22 11:00 AM dev branch 100% 100% .

100% 100% 100% 100% 100% 100% 100% 100% 100% 100% .

2025 5/30 11:00 AM nf-schema 100% 100% HealthOmics 100% 100% . 100%
nf-schema 100% 100% nf-validation 100% 100% . 100% 100% 100%
100% 100% 100% 100% . (100%)

????????? ?? ?? ??

Step1 - Ec2 ?? ?? ??

Ec2 100% 100% CloudShell 100% 100% 100% . (100%) 100% 100% .

- t2.medium
- Amazon linux 2023 AMI
- 64 bit x86

Specify instance type options

Launch an instance [Info](#)

Amazon EC2 allows you to create virtual machines, or instances, that run on the AWS Cloud. Quickly get started by following the simple steps below.

Name and tags [Info](#)

Name

aho-migration

[Add additional tags](#)

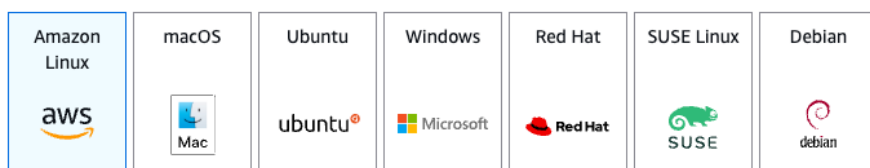
▼ Application and OS Images (Amazon Machine Image) [Info](#)

An AMI is a template that contains the software configuration (operating system, application server, and applications) required to launch your instance. Search or Browse for AMIs if you don't see what you are looking for below

Recents

My AMIs

Quick Start



[Browse more AMIs](#)

Including AMIs from AWS, Marketplace and the Community

Amazon Machine Image (AMI)

Amazon Linux 2023 AMI

ami-04999cd8f2624f834 (64-bit (x86), uefi-preferred) / ami-00565a15a71e4402a (64-bit (Arm), uefi)
Virtualization: hvm ENA enabled: true Root device type: ebs

Free tier eligible

Description

Amazon Linux 2023 is a modern, general purpose Linux-based OS that comes with 5 years of long term support. It is optimized for AWS and designed to provide a secure, stable and high-performance execution environment to develop and run your cloud applications.

Amazon Linux 2023 AMI 2023.7.20250512.0 x86_64 HVM kernel-6.1

Architecture

64-bit (x86)

Boot mode

uefi-preferred

AMI ID

ami-04999cd8f2624f834

Publish Date

2025-05-09

Username

ec2-user



Verified provider

▼ Instance type [Info](#) | [Get advice](#)

Instance type

t2.medium

Family: t2 2 vCPU 4 GiB Memory Current generation: true On-Demand SUSE base pricing: 0.1464 USD per Hour
On-Demand Ubuntu Pro base pricing: 0.0499 USD per Hour On-Demand Linux base pricing: 0.0464 USD per Hour
On-Demand RHEL base pricing: 0.0752 USD per Hour On-Demand Windows base pricing: 0.0644 USD per Hour

☒ All generations

[Compare instance types](#)

[Additional costs apply for AMIs with pre-installed software](#)

Generate key pairs

If you don't already have a key pair then select create keypair

▼ Key pair (login) [Info](#)

You can use a key pair to securely connect to your instance. Ensure that you have access to the selected key pair before you launch the instance.

Key pair name - *required*

omics-tutorial



[Create new key pair](#)

Give the key pair a name and select the key pair type and format.

Security groups

Ensure that you have access to the instance via SSH.

▼ Network settings Info

Edit

Network | Info

vpc-006b1a16ae3e759c6 | default

Subnet | Info

No preference (Default subnet in any availability zone)

Auto-assign public IP | Info

Enable

Additional charges apply when outside of free tier allowance

Firewall (security groups) | Info

A security group is a set of firewall rules that control the traffic for your instance. Add rules to allow specific traffic to reach your instance.

☒ Create security group

☐ Select existing security group

We'll create a new security group called 'launch-wizard-2' with the following rules:

☒ Allow SSH traffic from

Helps you connect to your instance

Anywhere
0.0.0.0/0

☐ Allow HTTPS traffic from the internet

To set up an endpoint, for example when creating a web server

☐ Allow HTTP traffic from the internet

To set up an endpoint, for example when creating a web server

⚠ Rules with source of 0.0.0.0/0 allow all IP addresses to access your instance. We recommend setting security group rules to allow access from known IP addresses only.

×

Storage

Create a root volume of 20GB

▼ Configure storage Info

Advanced

1x 20 GiB gp3 Root volume, 3000 IOPS, Not encrypted

ℹ Free tier eligible customers can get up to 30 GB of EBS General Purpose (SSD) or Magnetic storage

×

Add new volume

🕒 Click refresh to view backup information

The tags that you assign determine whether the instance will be backed up by any Data Lifecycle Manager policies.

↻

0 x File systems


Edit

Instance Profile

We will create an IAM instance profile to allow administrator permissions for the EC2 instance, for the purpose of this workshop. However, in general, AWS security best practices recommendation is that you create an instance profile with least privilege. This configuration is not recommended in your own account, you should discuss organizational best-practices

with your IT team to configure appropriate permissions.

Advanced Details 

 **Advanced details** [Info](#)

IAM Instance profile  [Create new IAM profile](#) 

▼ **Advanced details** [Info](#)

Domain join directory | [Info](#)

Select

 [Create new directory](#) 

IAM instance profile | [Info](#)

Select

 [Create new IAM profile](#) 

AWS service, EC2 

Step 1

☒ Select trusted entity

Step 2

☐ Add permissions

Step 3

☐ Name, review, and create

Select trusted entity [Info](#)

Trusted entity type



AWS service

Allow AWS services like EC2, Lambda, or others to perform actions in this account.



AWS account

Allow entities in this account to perform actions in this account.



SAML 2.0 federation

Allow users federated with SAML 2.0 from a corporate directory to perform actions in this account.



Custom trust policy

Create a custom trust policy for this account.

Use case

Allow an AWS service like EC2, Lambda, or others to perform actions in this account.

Service or use case

EC2

Choose a use case for the specified service.

Use case



EC2

Allows EC2 instances to call AWS services on your behalf.



EC2 Role for AWS Systems Manager

Allows EC2 instances to call AWS services like CloudWatch and Systems Manager on your behalf.



EC2 Spot Fleet Role

Allows EC2 Spot Fleet to request and terminate Spot Instances on your behalf.



EC2 - Spot Fleet Auto Scaling

Allows Auto Scaling to access and update EC2 spot fleets on your behalf.



EC2 - Spot Fleet Tagging

Allows EC2 to launch spot instances and attach tags to the launched instances on your behalf.



EC2 - Spot Instances

Allows EC2 Spot Instances to launch and manage spot instances on your behalf.



EC2 - Spot Fleet

Allows EC2 Spot Fleet to launch and manage spot fleet instances on your behalf.



EC2 - Scheduled Instances

Allows EC2 Scheduled Instances to manage instances on your behalf.

Add Permissions ☐ ☐ ☐ ☐

AdministratorAccess ☐ (☐)

Step 1

● Select trusted entity

Step 2

● Add permissions

Step 3

○ Name, review, and create

Add permissions

Info

Permissions policies (1/1389)

Info

Choose one or more policies to attach to your new

Q Search

-

Policy name

☒

AdministratorAccess

☐

AdministratorAccess-Amplify

role name (EC2adminAccess)

- Step 1

● Select trusted entity
- Step 2

● Add permissions
- Step 3

● **Name, review, and create**

Name, review, and create

Role details

Role name
Enter a meaningful name to identify this role.

EC2adminAccess

Maximum 64 characters. Use alphanumeric and '+=, @-_' characters.

Description
Add a short explanation for this role.

Allows EC2 instances to call AWS services on your behalf.

Maximum 1000 characters. Use letters (A-Z and a-z), numbers (0-9), ta

Step 1: Select trusted entities

Trust policy

```
1 {  
2   "Version": "2012-10-17",  
3   "Statement": [  
4     {  
5       "Effect": "Allow",  
6       "Action": [  
7         "sts:AssumeRole"  
8       ],  
9       "Principal": {  
10        "Service": [  
11          "ec2.amazonaws.com"  
12        ]  
13      }  
14    }  
15  ]  
16 }
```

EC2 instance

EC2adminAccess

IAM instance profile

Info

EC2adminAccess

arn:aws:iam::instance-profile/EC2adminAccess



User Data

□ □□ **User data** □□ □□ □□

```
#!/bin/bash
```

```
echo "Setting up NodeJS Environment"
```

```
curl https://raw.githubusercontent.com/nvm-sh/nvm/v0.34.0/install.sh | bash
```

```
echo 'export NVM_DIR="/.nvm"' >> /home/ec2-user/.bashrc
```

```
echo '[ -s "$NVM_DIR/nvm.sh" ] && . "$NVM_DIR/nvm.sh" # This loads nvm' >> /home/ec2-user/.bashrc
```

```
# Dot source the files to ensure that variables are available within the current shell
```

```
. /.nvm/nvm.sh
```

```
. ~/.bashrc
```

```
nvm install --lts
```

```
python3 -m ensurepip --upgrade
```

```
pip3 install boto3
```

```
npm install -g aws-cdk
```

```
sudo yum -y install git
```


User data - optional | [Info](#)

Upload a file with your user data or enter it in the field.

 [Choose file](#)

```
#!/bin/bash

echo "Setting up NodeJS Environment"
curl https://raw.githubusercontent.com/nvm-sh/nvm/v0.34.0/install.sh | bash

echo 'export NVM_DIR="/.nvm"' >> /home/ec2-user/.bashrc
echo '[ -s "$NVM_DIR/nvm.sh" ] && . "$NVM_DIR/nvm.sh" # This loads nvm' >> /home/ec2-user/.bashrc

# Dot source the files to ensure that variables are available within the current shell
. /.nvm/nvm.sh
. ~/.bashrc
nvm install --lts

python3 -m ensurepip --upgrade
```

☐ User data has already been base64 encoded

Launch instance   

▼ Summary

Number of instances | [Info](#)

1

Software Image (AMI)

Amazon Linux 2023 AMI 2023.7.2...[read more](#)

ami-l

Virtual server type (instance type)

t2.medium

Firewall (security group)

launch-wizard-1

Storage (volumes)

1 volume(s) - 20 GiB

Free tier: In your first year of opening an AWS account, you get 750 hours per month of t2.micro instance usage (or t3.micro where t2.micro isn't available) when used with free tier AMIs, 750 hours per month of public IPv4 address usage, 30 GiB of EBS storage, 2 million I/Os, 1 GB of snapshots, and 100 GB of bandwidth to the internet.

[Cancel](#)

[Launch instance](#)

[Preview code](#)

Step 2 - workflow migration code? ????

SSH ☐ , ☐ ☐ console ☐ ☐ ☐ ☐ .

Instances (1/1)

Info

Find Instance by attribute or tag (case-sensitive)

All states

Instance ID = i-

X

Clear filters

Last updated less than a minute ago

Refresh

Connect

<input checked="" type="checkbox"/>	Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public IPv4 DNS
<input checked="" type="checkbox"/>	aho-migration	i-	Running	t2.medium	Initializing	View alarms +	us-west-2a	.us...

Connect to instance [Info](#)

Connect to your instance i-00b08bcd189b57c0 (aho-migration) using any of these options


EC2 Instance Connect



Session Manager

SSH client


EC2 serial console

Instance ID

 i- (aho-migration)

1. Open an SSH client.
2. Locate your private key file. The key used to launch this instance is omics-tutorial.pem
3. Run this command, if necessary, to ensure your key is not publicly viewable.
 `chmod 400 "omics-tutorial.pem"`
4. Connect to your instance using its Public DNS:
 .us-west-2.compute.amazonaws.com

Example:

 `ssh -i "omics-tutorial.pem" ec2-user@.compute.amazonaws.com`

 **Note:** In most cases, the guessed username is correct. However, read your AMI usage instructions to

Use SSH to access the instance, you will need to ensure the key you created earlier is in the `.ssh` folder in your home directory and has the correct permissions(`chmod 400`). You will **need replace** `ec2-user@10.11.12.123` with the IP address of your instance. This can be found in the EC2 management console.

```
ssh -i .ssh/omics-tutorial.pem ec2-user@10.11.12.123
```

Get the region using the `ec2-metadata` command. This handy oneliner stores it as a variable

```
REGION=$(ec2-metadata --availability-zone | sed 's/placement: \(.*\).$/\1/')
```

Get the account number (if you don't already know it)

```
ACCOUNT_NUMBER=$(aws sts get-caller-identity --query 'Account' --output text)
```

Bootstrapping

Now you can bootstrap cdk replacing `ACCOUNT-NUMBER` with your account number nad using the `$REGION` variable created earlier.

```
cdk bootstrap aws://$ACCOUNT_NUMBER/$REGION
```

Download the workflow migration code

This step downloads the code which is used to migrate the workflow into AWS HealthOmics Workflows

```
cd ~  
git clone https://github.com/aws-samples/amazon-ecr-helper-for-aws-healthomics.git  
  
git clone https://github.com/aws-samples/aws-healthomics-tutorials.git  
  
cd amazon-ecr-helper-for-aws-healthomics
```

Install and deploy the code.

```
npm install
cdk deploy --all
```


???? ??

nf-core HealthOmics private workflow

- Cloning a workflow from nf-core
- Generating artifacts for migration
- Privatizing containers by migrating them to your private Amazon ECR

   **Bash**  

```
export yourbucket="your-bucket-name"
export your_account_id="your-account-id" #ACCOUNT_NUMBER
export region="your-region"
export workflow_name="your-workflow-name"
export omics_role_name="your_omics_rolename"

# if not exist the bucket, let's create.
#aws s3 mb $yourbucket
```

Clone nf-core repository that you want to migrate

```
cd ~
git clone https://github.com/nf-core/spatialvi.git
```

Docker Image Manifest? ??

```
cp /home/workshop/amazon-ecr-helper-for-aws-healthomics/lib/lambda/parse-image-
uri/public_registry_properties.json namespace.config
```

```
namespace.config
```

```
[ec2-user@ip-172-31-44-194 ~]$ cat namespace.config
{
  "public.ecr.aws": { "namespace": "ecr-public", "pull_through": true },
  "ecr-public": { "namespace": "ecr-public", "pull_through": true },

  "quay.io": { "namespace": "quay", "pull_through": true },
  "quay": { "namespace": "quay", "pull_through": true },

  "us.gcr.io": { "namespace": "gcr", "pull_through": false },
  "eu.gcr.io": { "namespace": "gcr", "pull_through": false },
  "asia.gcr.io": { "namespace": "gcr", "pull_through": false },
  "pkg.dev": { "namespace": "gar", "pull_through": false },
  "nvcr.io": { "namespace": "nvcr", "pull_through": false },

  "ghcr.io": { "namespace": "ghcr", "pull_through": false },
  "mcr.microsoft.com": { "namespace": "mcr", "pull_through": false },

  "dockerhub": { "namespace": "dockerhub", "pull_through": false },
  "": { "namespace": "dockerhub", "pull_through": false }
}[ec2-user@ip-172-31-44-194 ~]$
```

This file will be used as one of the inputs for the `inspect_nf.py` script. This script parses the entire nextflow workflow project and identifies all public docker images across all the tools and generates a manifest which will be passed as an input to the migration tool.

```
inspect_nf.py
```

```
python3 aws-healthomics-tutorials/utils/scripts/inspect_nf.py \
--output-manifest-file spatialvi_dev_docker_images_manifest.json \
-n namespace.config \
--output-config-file omics.config \
--region $region \
~/spatialvi/
```

```
spatialvi_dev_docker_images_manifest.json omics.config
```

```
spatialvi_dev_docker_images_manifest.json
```

```
[ec2-user@ip-172-31-44-194 ~]$ cat spatialvi_dev_docker_images_manifest.json
{
  "manifest": [
    "quay.io/biocontainers/fastqc:0.12.1--hdfd78af_0",
    "quay.io/docker.io/erikfas/spatialvi",
    "quay.io/docker.io/multiqc/multiqc:v1.24.1",
    "quay.io/nf-core/spaceranger:3.1.3",
    "quay.io/nf-core/ubuntu:22.04"
  ]
}
```

We can find the Docker image for spaceranger on Docker Hub. (Docker container images are stored on Docker Hub or other registries). We can find the Docker image for spaceranger on Docker Hub. The Docker image for spaceranger is located at <https://hub.docker.com/search?q=spaceranger>.

```
{
  "manifest": [
    "quay.io/biocontainers/fastqc:0.12.1--hdfd78af_0",
    "erikfas/spatialvi:latest",
    "multiqc/multiqc:v1.24.1",
    "cumulusprod/spaceranger:3.1.3",
    "ubuntu:22.04"
  ]
}
```

The omics.config output file is also generated, which needs to be added to the nextflow workflow project. The purpose of this config is to inform nextflow to use the ECR docker image locations instead of the ones specified in the nextflow project code. The omics.config should look like this:

```
[ec2-user@ip-172-31-44-194 ~]$ cat omics.config
params {
  ecr_registry = '██████████.dkr.ecr.us-west-2.amazonaws.com'
  outdir = '/mnt/workflow/pubdir'
}

manifest {
  nextflowVersion = '!=22.04.0'
}

conda {
  enabled = false
}

docker {
  enabled = true
  registry = params.ecr_registry
}

process {
  withName: '.*' { conda = null }
  withName: '(.+)?MERGE_SDATA' { container = 'quay/docker.io/erikfas/spatialvi' }
  withName: '(.+)?READ_DATA' { container = 'quay/docker.io/erikfas/spatialvi' }
  withName: '(.+)?FASTQC' { container = 'quay/biocontainers/fastqc:0.12.1--hdfd78af_0' }
  withName: '(.+)?MULTIQC' { container = 'quay/docker.io/multiqc/multiqc:v1.24.1' }
  withName: '(.+)?QUARTONOTEBOOK' { container = 'quay/docker.io/erikfas/spatialvi' }
  withName: '(.+)?SPACERANGER_COUNT' { container = 'quay/nf-core/spaceranger:3.1.3' }
  withName: '(.+)?UNTAR' { container = 'quay/nf-core/ubuntu:22.04' }
}
[ec2-user@ip-172-31-44-194 ~]$ █
```

omics.config process .

```
process {
  withName: '.*' { conda = null }
  withName: '(.+)?MERGE_SDATA' { container = 'erikfas/spatialvi' }
  withName: '(.+)?READ_DATA' { container = 'erikfas/spatialvi' }
  withName: '(.+)?FASTQC' { container = 'quay/biocontainers/fastqc:0.12.1--hdfd78af_0' }
  withName: '(.+)?MULTIQC' { container = 'multiqc/multiqc:v1.24.1' }
  withName: '(.+)?QUARTONOTEBOOK' { container = 'erikfas/spatialvi' }
  withName: '(.+)?SPACERANGER_COUNT' { container = 'quay/nf-core/spaceranger:3.1.3' }
  withName: '(.+)?UNTAR' { container = 'quay/nf-core/ubuntu:22.04' }
}
```

???? ??? (into Amazon ECR)

```
aws stepfunctions start-execution \
--state-machine-arn arn:aws:states:$region:$your_account_id:stateMachine:omx-container-puller \
```

```
--input file://spatialvi_dev_docker_images_manifest.json
```

step function ☐ ☐ state machines ☐ omx-container-puller ☐ ☐ ☐ Execution ☐ ☐ ☐ ☐ ☐

Step Functions

State machines

Activities

Developer resources

Online learning workshop

Local Development

Data flow simulator

Feature spotlight

Documentation

Join our feedback panel

Updated extension for VS Code

With the latest AWS Toolkit extension for Visual Studio Code, you can edit and test workflows graphically with Workflow Studio. [Learn more](#)

Download the AWS Toolkit

State machines (2)

View execution counts

View details

Edit

Copy to new

Delete

Search for state machines

Any type

	Name	Type	Creation date	Status
<input type="radio"/>	omx-container-builder	Standard	Apr 15, 2025, 15:44:36 (UTC+09:00)	Active
<input type="radio"/>	omx-container-puller	Standard	Apr 15, 2025, 14:55:08 (UTC+09:00)	Active

omx-container-puller

Details

Arn

[arn:aws:states:::states:::stateMachine:omx-container-puller](#)

IAM role ARN

[arn:aws:iam:::role/OmxEcrHelper-ContainerPull-StateMachineRoleB840431D-av8QNi1KB4Zr](#)

Type

Standard

Status

Active

Creation date

Apr 18, 2025, 13:00:17 (UTC+09:00)

X-Ray tracing

Disabled

Executions

Monitoring

Logging

Definition

Aliases

Versions

Tags

Executions (0/4)

Filter executions by property or value

All

Last 15 months

Local timezone

[234f30d5-026d-467f-a557-c205a5e778b0](#)

Clear filters

<input type="checkbox"/>	Name	Status	Start Time (local)	End Time (local)	Duration
<input type="checkbox"/>	234f30d5-026d-467f-a557-c205a5e778b0	Succeeded	May 8, 2025, 18:16:18	May 8, 2025, 18:21:09	00:04:50.895

nf-core project ?? ?????

To use the newly migrated containers for our workflow, copy the omics.config generated from inspect_nf.py in the /conf dir of the spatialvi project.

```
mv omics.config spatialvi/conf
```

Update the nextflow.config file in the root dir of the scrnaseq project by adding to the bottom of the file:


```
echo "includeConfig 'conf/omics.config'" >> spatialvi/nextflow.config
```

?? HealthOmics ?????? ???

??1. ????? ??

parameter-description.json                                 .

```
cat > parameter-description.json <<EOF
{
  "input":{
    "description":"CSV samplesheet containing sample information",
    "optional":false
  },
  "spaceranger_probeset":{
    "description":"Probe set file for Space Ranger analysis",
    "optional":false
  },
  "spaceranger_reference":{
    "description":"Reference genome tarball for Space Ranger",
    "optional":false
  },
  "qc_min_counts":{
    "description":"Minimum count threshold for QC filtering",
    "optional":true
  },
  "qc_min_genes":{
    "description":"Minimum number of genes threshold for QC filtering",
    "optional":true
  }
}
EOF
```

??2. ?????? ?????

```
zip -r spatialvi.zip spatialvi/ -x "*\.*" "*\.*/*"
```

```
aws omics create-workflow \
  --name ${workflow_name} \
  --definition-uri s3://${yourbucket}/workshop/spatialvi-workflow.zip \
  --parameter-template file://parameter-description.json \
  --engine NEXTFLOW
```

??3. ???? ? ?

```
workflow_id=$(aws omics list-workflows --name ${workflow_name} --query 'items[0].id' --output text)
echo $workflow_id
```

????? ?????

???? ??

parameter-description.json input.json .

 . ()

□□ □□□□ □□□ □□ □□□□ □□ □ □□□□ .

```
wget "https://raw.githubusercontent.com/nf-core/test-datasets/spatialvi/testdata/human-brain-cancer-11-
mm-capture-area-ffpe-2-standard_v2_ffpe_cytassist/samplesheet_spaceranger.csv"
wget "https://raw.githubusercontent.com/nf-core/test-datasets/spatialvi/testdata/human-brain-cancer-11-
mm-capture-area-ffpe-2-standard_v2_ffpe_cytassist/outs/probe_set.csv"
wget "https://raw.githubusercontent.com/nf-core/test-
datasets/spatialvi/testdata/homo_sapiens_chr22_reference.tar.gz"
```

```
aws s3 mv samplesheet_spaceranger.csv s3://${yourbucket}/spatialvi/
aws s3 mv probe_set.csv s3://${yourbucket}/spatialvi/
```


```
aws s3 mv homo_sapiens_chr22_reference.tar.gz s3://${yourbucket}/spatialvi/
```












sample sheet

```
cat << EOF > samplesheet_spaceranger.csv
sample,fastq_dir,cytainage,slide,area,manual_alignment,slidefile
CytAssist_11mm_FFPE_Human_Glioblastoma_2,s3://${bucket}/spatialvi/fastq_dir.tar.gz,s3://${bucket}/spatialvi/
CytAssist_11mm_FFPE_Human_Glioblastoma_image.tif,V52Y10-317,B1,,s3://${bucket}/spatialvi/V52Y10-317.gpr
EOF
```

  **samplesheet**  **s3** 

```
aws s3 cp samplesheet_spaceranger.csv s3://${yourbucket}/spatialvi/samplesheet_spaceranger.csv
```

 **json**  ( **sample sheet**   **input**         )


    **probe_set.csv, homo_sapiens_chr22_reference.tar.gz**        .

```
cat > input.json <<EOF
{
  "input": "s3://${yourbucket}/spatialvi/samplesheet_spaceranger.csv",
  "spaceranger_probeset": "s3://${yourbucket}/spatialvi/probe_set.csv",
  "spaceranger_reference": "s3://${yourbucket}/spatialvi/homo_sapiens_chr22_reference.tar.gz",
  "qc_min_counts": 5,
  "qc_min_genes": 3
}
EOF
```

Policy ??

Prepare IAM service role to run AWS HealthOmics workflow

your-bucket-name, your-account-id, your-region          .

omics_workflow_policy.json 

```
cat << EOF > omics_workflow_policy.json
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "s3:GetObject"
      ],
      "Resource": [
        "arn:aws:s3:::${yourbucket}/*",
        "arn:aws:s3:::aws-genomics-static-${region}/workflow_migration_workshop/nfcore-scrnaseq-v2.3.0/*"
      ]
    },
    {
      "Effect": "Allow",
      "Action": [
        "s3:ListBucket"
      ],
      "Resource": [
        "arn:aws:s3:::${yourbucket}",
        "arn:aws:s3:::aws-genomics-static-${region}/workflow_migration_workshop/nfcore-scrnaseq-v2.3.0",
        "arn:aws:s3:::aws-genomics-static-${region}/workflow_migration_workshop/nfcore-scrnaseq-v2.3.0/*"
      ]
    },
    {
      "Effect": "Allow",
      "Action": [
        "s3:PutObject"
      ],
      "Resource": [
        "arn:aws:s3:::${yourbucket}/*"
      ]
    },
    {
      "Effect": "Allow",
      "Action": [
        "logs:DescribeLogStreams",
        "logs:CreateLogStream",
```

```

        "logs:PutLogEvents"
    ],
    "Resource": [
        "arn:aws:logs:${region}:${your_account_id}:log-group:/aws/omics/WorkflowLog:log-stream:*"
    ]
},
{
    "Effect": "Allow",
    "Action": [
        "logs:CreateLogGroup"
    ],
    "Resource": [
        "arn:aws:logs:${region}:${your_account_id}:log-group:/aws/omics/WorkflowLog:*"
    ]
},
{
    "Effect": "Allow",
    "Action": [
        "ecr:BatchGetImage",
        "ecr:GetDownloadUrlForLayer",
        "ecr:BatchCheckLayerAvailability"
    ],
    "Resource": [
        "arn:aws:ecr:${region}:${your_account_id}:repository/*"
    ]
}
]
}
EOF

```

```
echo "omics_workflow_policy.json" > omics_workflow_policy.json
```

```
trust_policy.json
```

```

cat << EOF > trust_policy.json
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",

```

```

    "Principal": {
        "Service": "omics.amazonaws.com"
    },
    "Action": "sts:AssumeRole",
    "Condition": {
        "StringEquals": {
            "aws:SourceAccount": "${your_account_id}"
        },
        "ArnLike": {
            "aws:SourceArn": "arn:aws:omics:${region}:${your_account_id}:run/*"
        }
    }
}
]
}
EOF

```

```
echo "trust_policy.json" | cat /dev/null > trust_policy.json
```

IAM Role ??

```
aws iam create-role --role-name ${omics_role_name} --assume-role-policy-document file://trust_policy.json
```

Policy document

```
aws iam put-role-policy --role-name ${omics_role_name} --policy-name OmicsWorkflowV1 --policy-document
file://omics_workflow_policy.json
```

????? ??

```

input.json, input_full.json

```

```

aws omics start-run \
  --name spatialvi_test_run_1 \
  --role-arn arn:aws:iam::${your_account_id}:role/${omics_role_name} \
  --workflow-id ${workflow_id} \
  --parameters file://input.json \
  --output-uri s3://${yourbucket}/output/

```



- <https://catalog.us-east-1.prod.workshops.aws/workshops/76d4a4ff-fe6f-436a-a1c2-f7ce44bc5d17/en-US/workshop/project-setup>

Revision #8

Created 22 May 2025 01:00:42 by Hyunmin Kim

Updated 26 May 2025 06:55:30 by Hyunmin Kim