Running NVIDIA Parabricks on AWS

Table of contents

List of Figures

Figure 19. Image Result

This guide shows how to run Parabricks on AWS HealthOmics and is divided into two parts:

The first part shows how to run Parabricks workflows using an EC2 instance. For this method, we will spin up a machine instance on AWS, pull the Parabricks container directly from NVIDIA, and run an example dataset. This option allows for the most flexibility in terms of Parabricks functionality and for easily integrating into larger pipelines and other AWS services.

The second part shows how to run Parabricks workflows using Amazon HealthOmics. This is Amazon's platform for bioinformatics research and allows you to store data, run analysis pipelines, and look at the results, all in one place. There are two ways to use HealthOmics. Ready2Run workflows are pre-made analysis pipelines where users can click on a pipeline they want to run, click on the data they want to use, and click run all without ever leaving the console GUI. Power users can also use the AWS CLI to start these jobs. The other way to run HealthOmics is through Private Workflows. These are great for if you want a little more control over the workflows and want to make edits to fit your needs exactly.

What is NVIDIA Parabricks?

Parabricks is an accelerated compute framework that supports applications across the genomics industry, primarily supporting analytical workflows for DNA, RNA, and somatic mutation detection applications. With industry leading compute times, Parabricks rapidly converts a FASTQ file to a VCF using multiple, industry validated variant callers and also includes the ability to QC and annotate those variants. As Parabricks is based upon publicly available tools, results are easy to verify and combine with other publicly available data sets.

More information is available on the Parabricks Product Page.

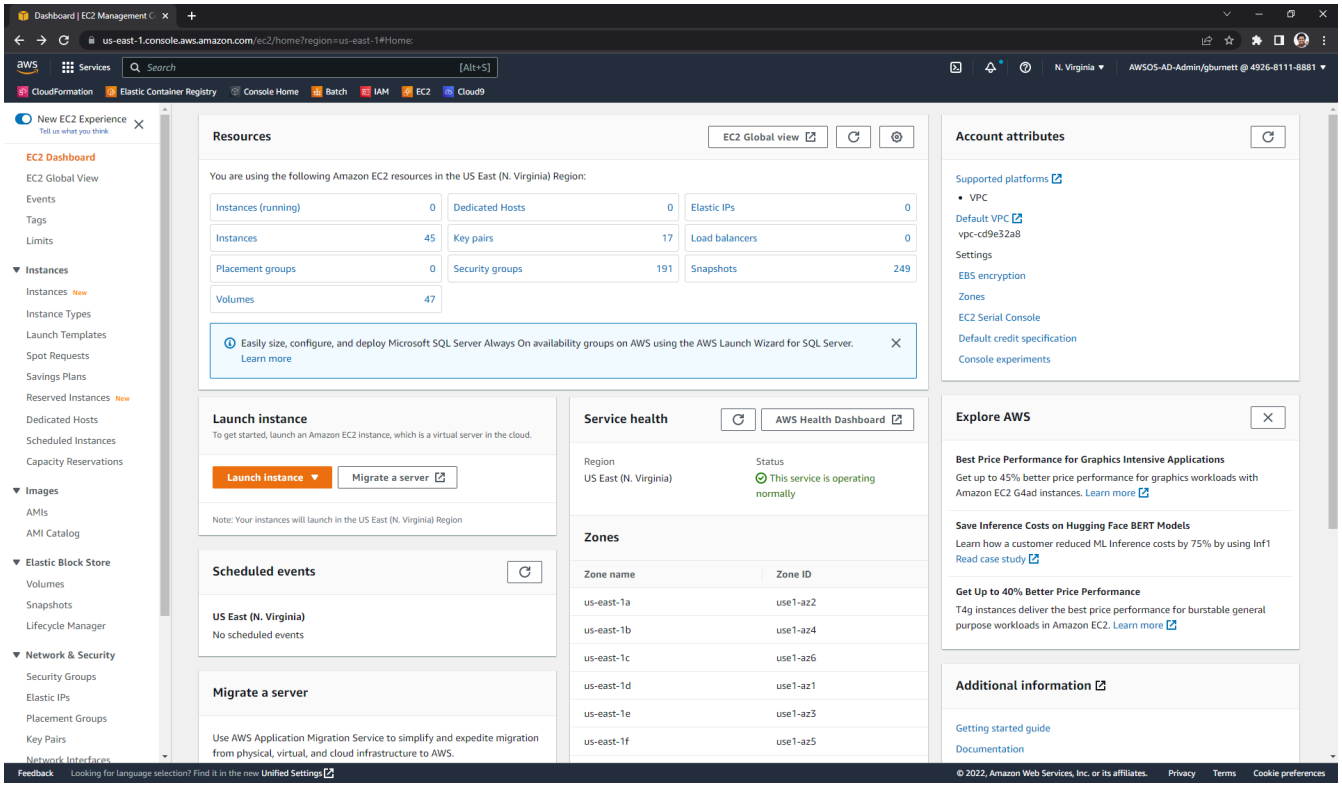
Detailed installation, usage, and tuning information is available in the Parabricks user guide.

Running Workflows on an EC2 Instance

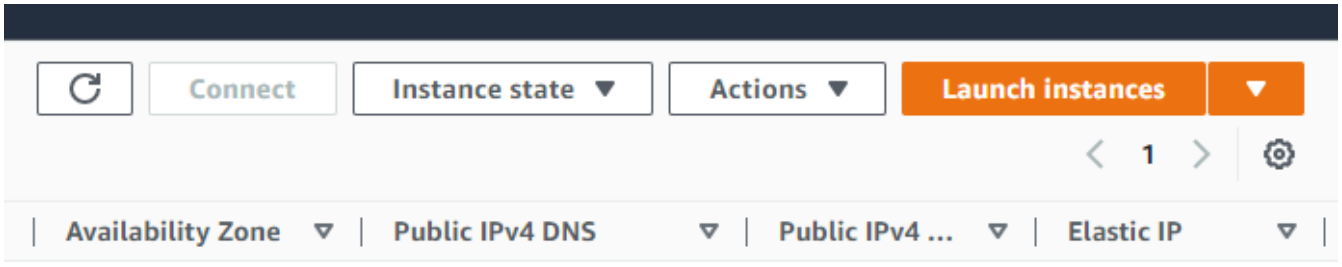
Starting an EC2 Instance

In this section, we will show how to start an EC2 instance on AWS.

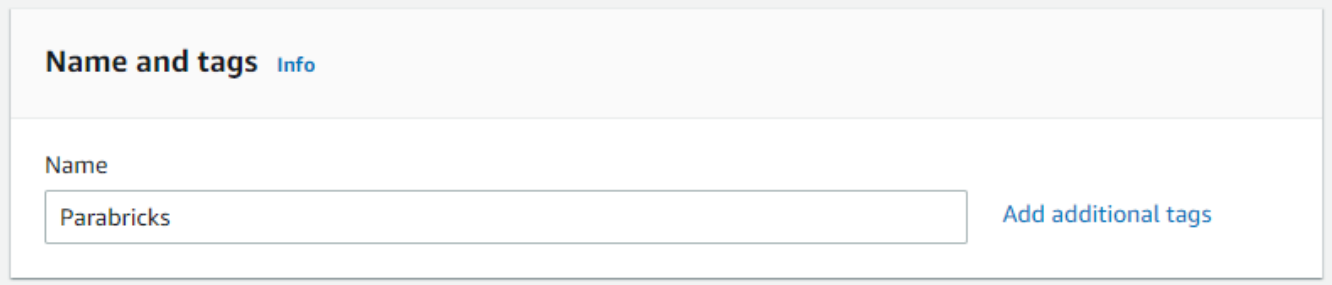
Begin by navigating to the EC2 console on AWS. The page should look something like this:



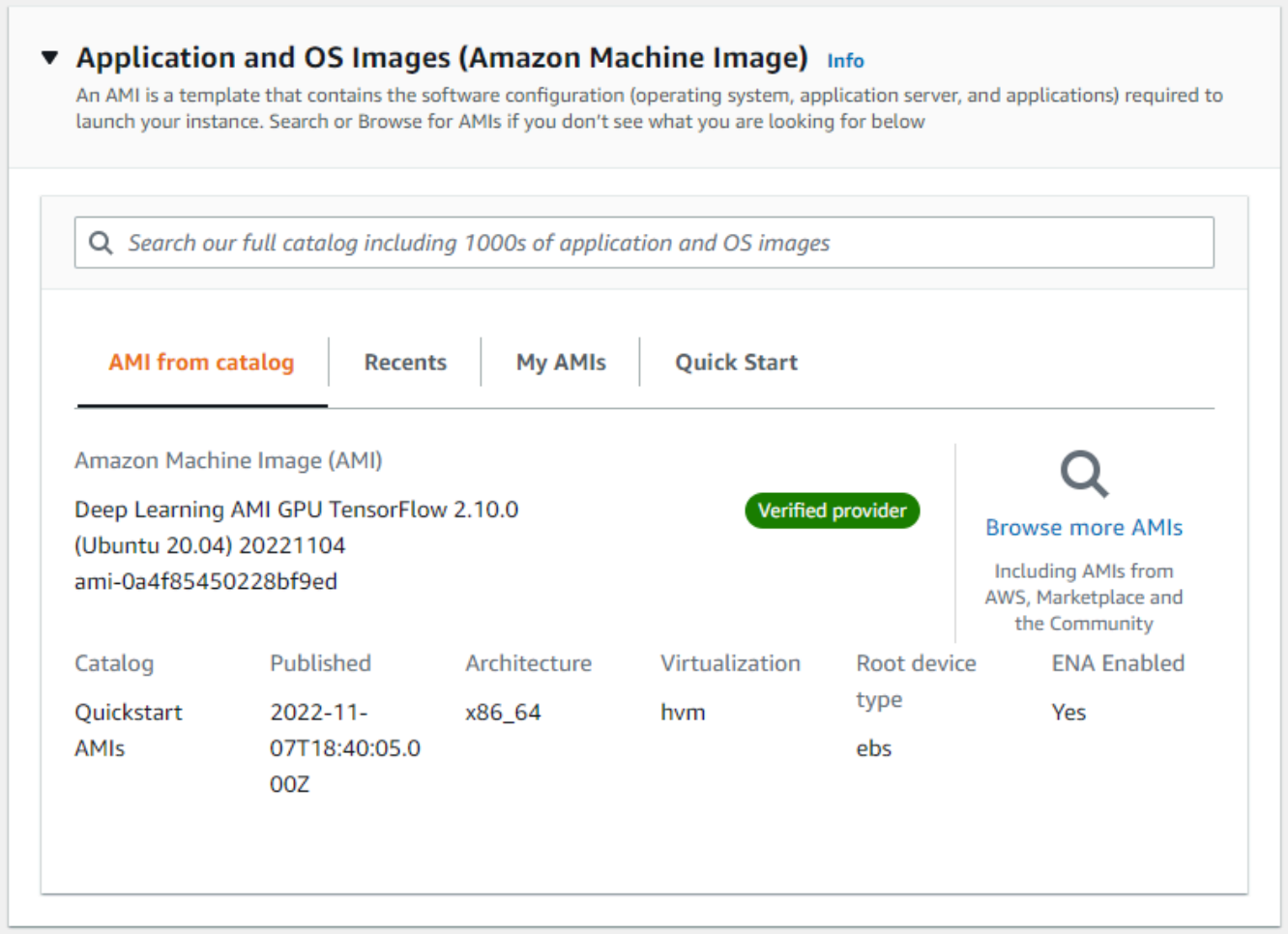
In the left sidebar under “Instances” click “Instances”. Here we can see all the instances we have created. Let’s create a new one where we will install Parabricks, by clicking “Launch instances” in the top right.



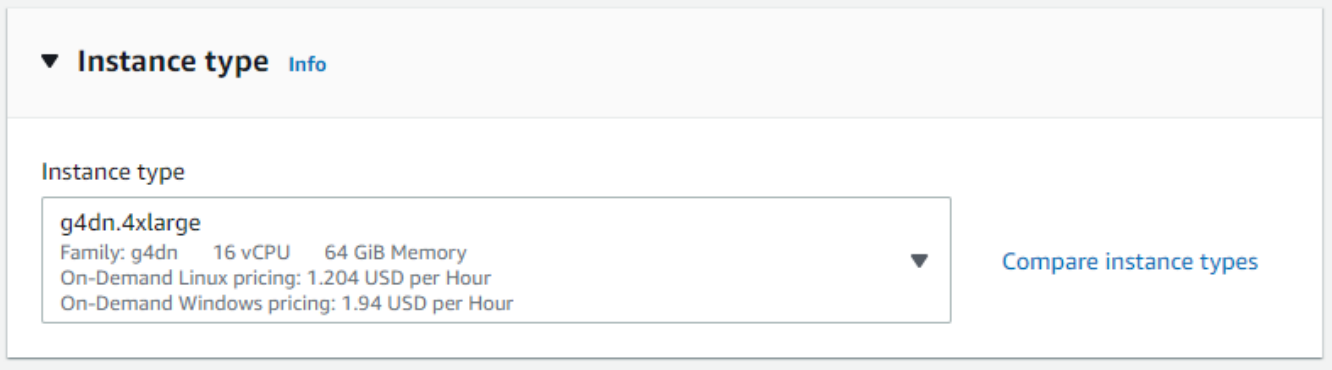
In this guide, we will name our instance “Parabricks” but it can be named anything.



We will use an Amazon Machine Image (AMI) that has all the software requirements for Parabricks. Under “Application and OS Images” search “Deep Learning AMI” and select any recent version.

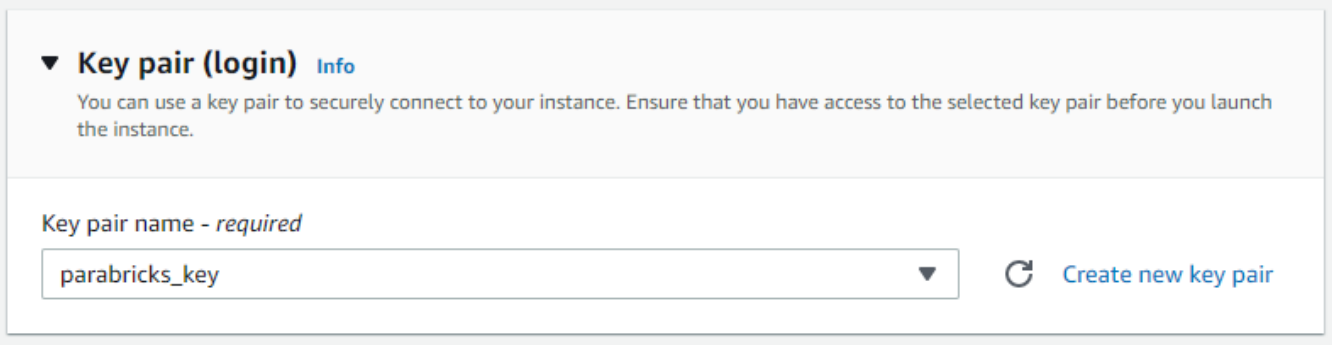


For installing and testing Parabricks, we will need an instance with at least 1 GPU. Under “Instance type” select “Compare instance types”. In the search bar type “g4dn.4xlarge” and select that instance type from the list of options. This instance has 1 NVIDIA T4 GPU with 16 vCPUs and 64 GB of RAM. Read more about g4dn instances on the AWS documentation.

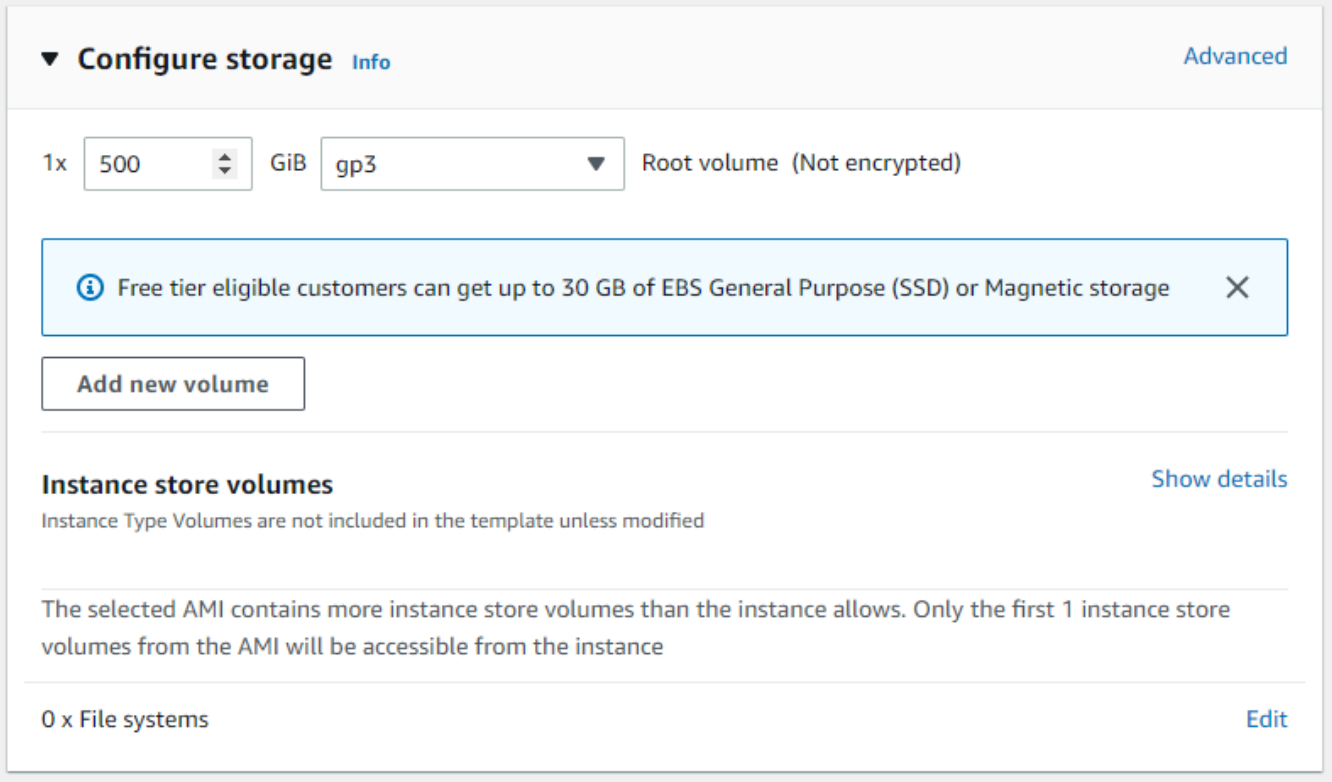


We need to select a key pair if we want to use SSH to log into the instance. For this tutorial, we will be logging into the instance using “EC2 Instance Connect” which does not require a key pair. In the “Key pair” drop-down, we will select the first options “Proceed without a key pair”.

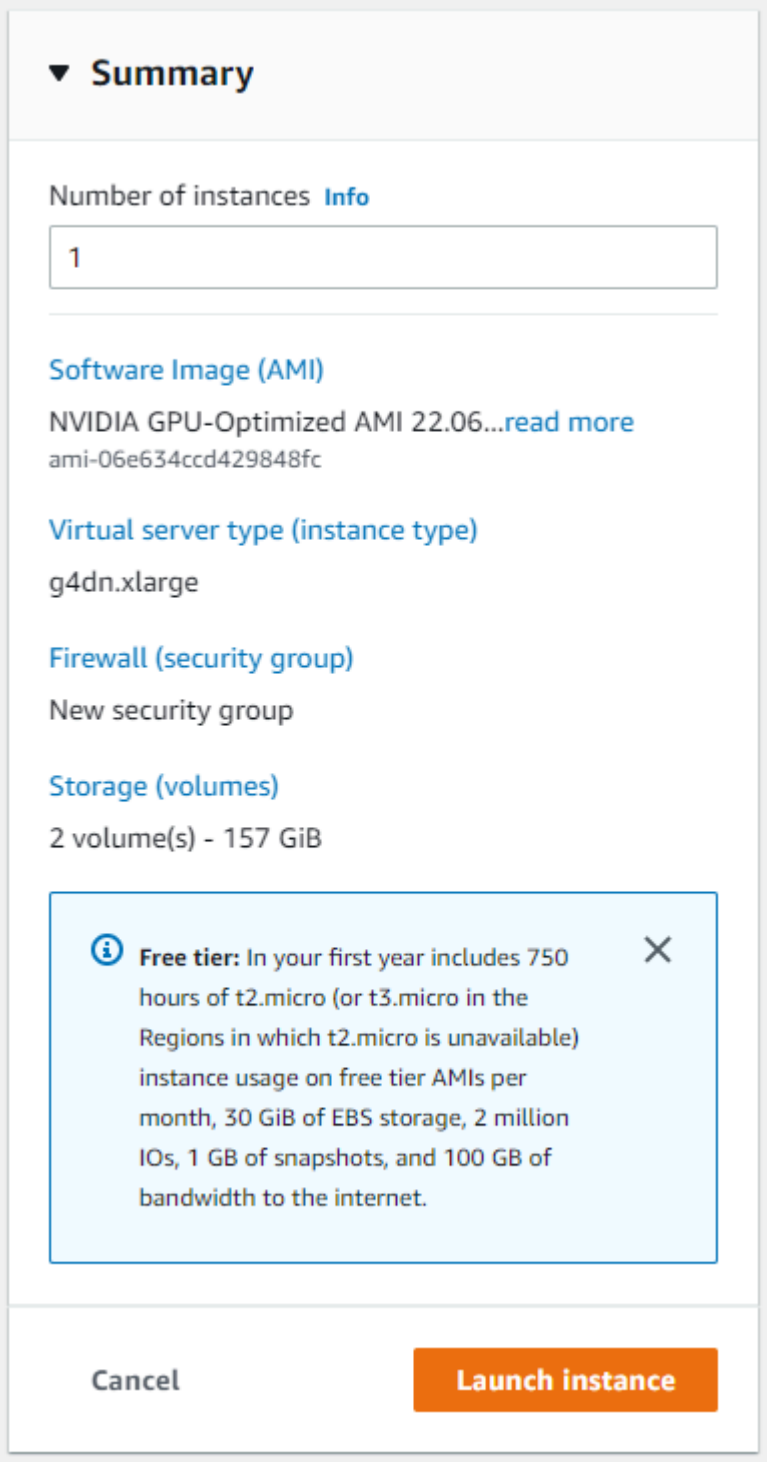
However, if you do want to generate a key pair, select “Create new key pair”, give the key pair a name, and select “Create key pair”. The key will automatically download. Save this for a later step.



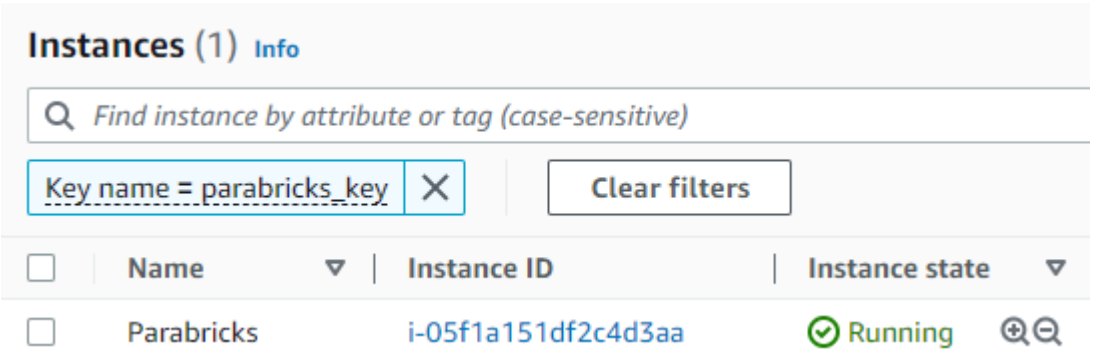
Lastly, we must increase the storage quota so that when we download and run our test data, we have enough disk space. Under “Configure storage” change the default Root volume size to 500 GB.



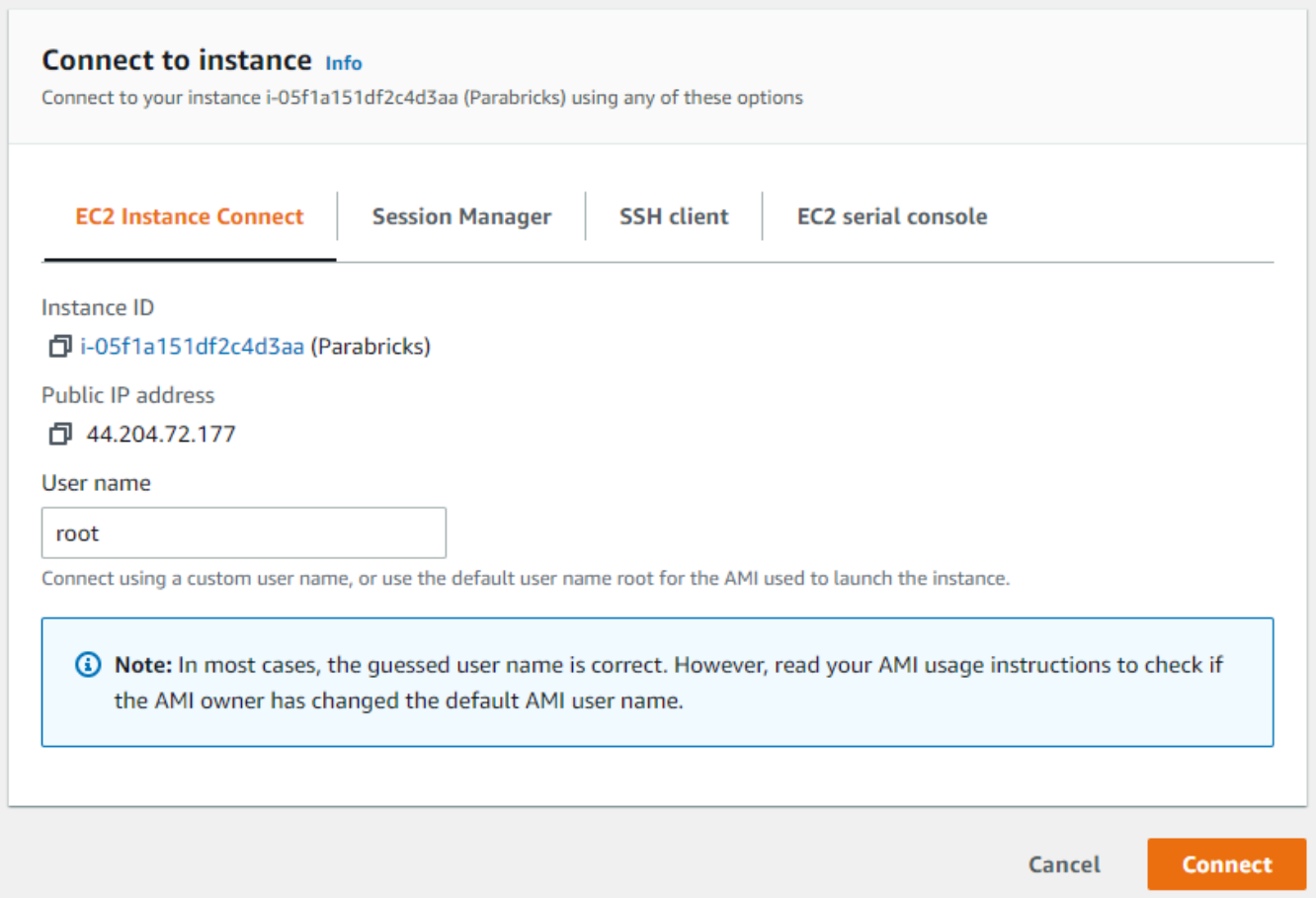
Our instance is ready to be launched now. Select “Launch instance”.



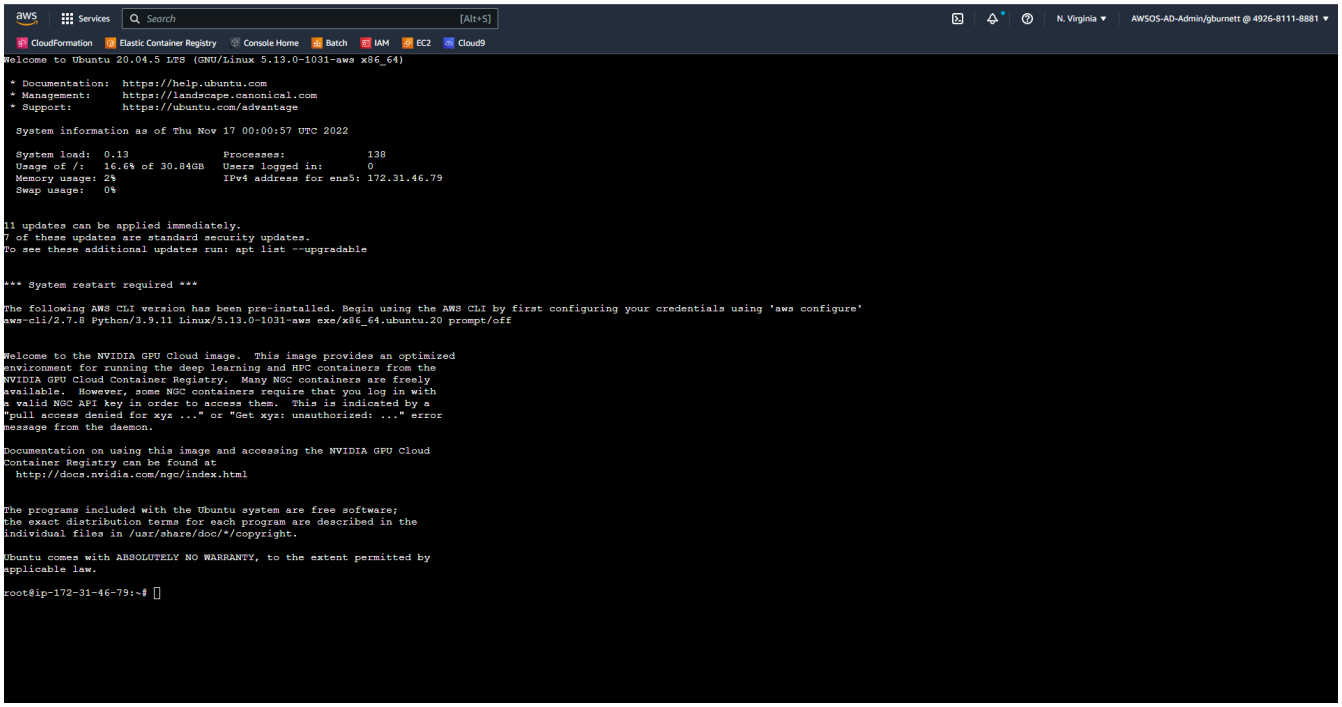
The instance should begin to launch. Navigate back to the “Instances” section of the left side panel and select “Instances to confirm that the instance is running.



Click on the checkbox next to the instance and a box will appear in the top right saying “Connect”. Click that button. If you generated a key pair in the previous steps, you can use it to connect using the SSH client. However, we will be connecting using “EC2 Instance Connect” which does not require a key-pair. Click connect.



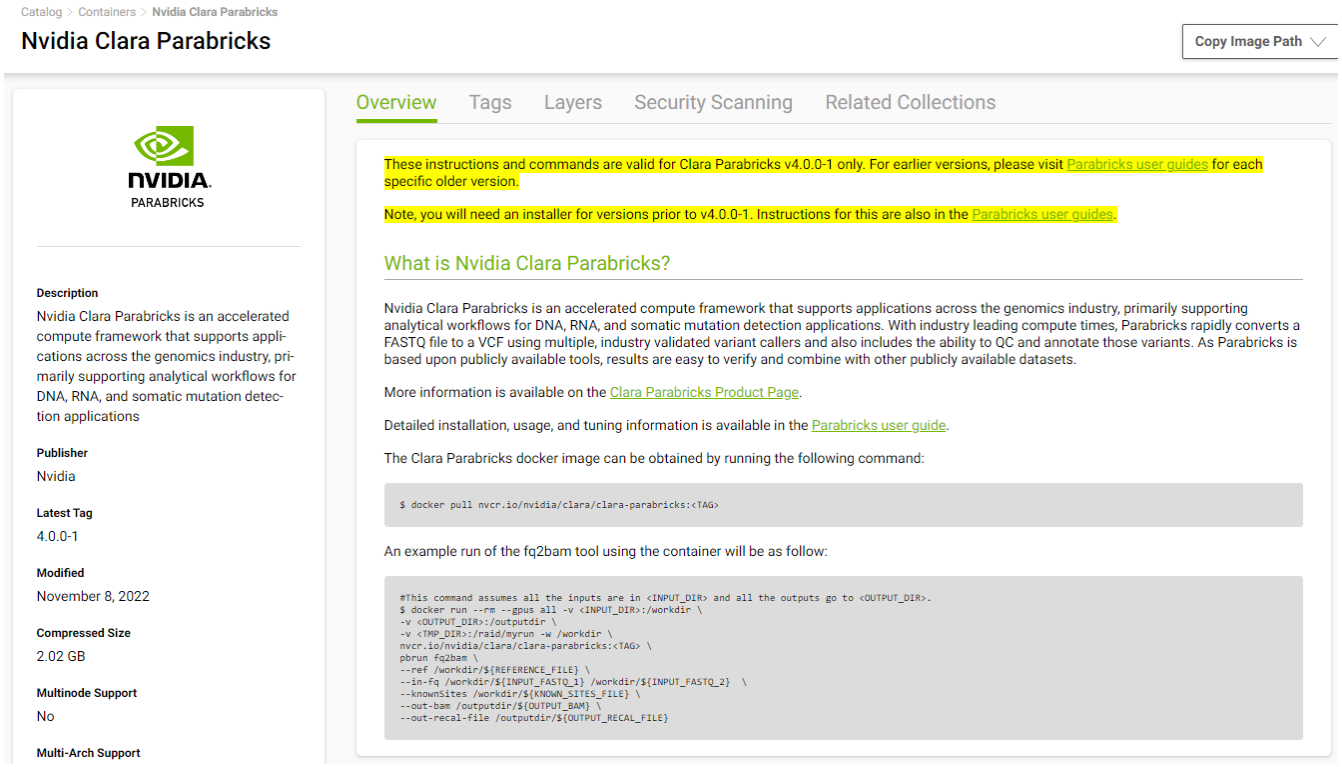
We are now greeted with a full terminal with our NVIDIA GPU-Optimized AMI pre-installed. We are now ready to start installing Parabricks.



Installing Parabricks

We will install Parabricks into our instance that we just created. To do this, we will use the NVIDIA GPU Cloud (NGC) to download the Parabricks Docker image.

Visit the Parabricks page on NGC to get the Docker pull command for the latest version of Parabricks.



Back in our EC2 instance, let's run the docker pull command:

```
$ docker pull nvcr.io/nvidia/clara/clara-parabricks:4.3.1-1
```

```
gburnett_nvidia_com@parabricks:~$ docker pull nvcr.io/nvidia/clara/clara-parabricks:4.0.0-1
4.0.0-1: Pulling from nvidia/clara/clara-parabricks
d7bfe07ed847: Pull complete
bbbbd451a669: Pull complete
773163705c35: Pull complete
d6949fcf1aef: Pull complete
3eb73064088b: Pull complete
a3ac3ab0ee35: Pull complete
8d88682a5e1d: Pull complete
Digest: sha256:0170beef24131a23bb63bc36ec059e493df1f04a4a78f9d2c5df9bce1d5d9a35
Status: Downloaded newer image for nvcr.io/nvidia/clara/clara-parabricks:4.0.0-1
nvcr.io/nvidia/clara/clara-parabricks:4.0.0-1
```

Now Parabricks is installed! Let's run some sample data to test it.

Testing Parabricks

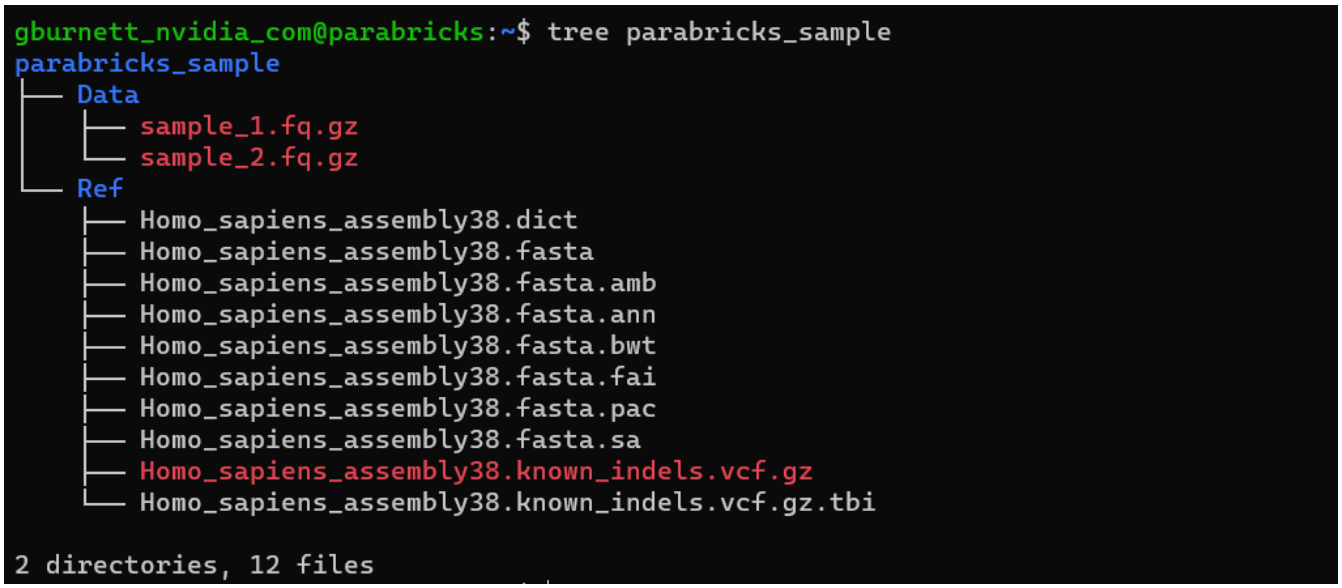
Parabricks provides a small sample dataset as a test for the installation and hardware which can be downloaded using:

```
wget -O parabricks_sample.tar.gz
"https://s3.amazonaws.com/parabricks.sample/parabricks_sample.tar.gz"
```

When the download completes we can untar the data using:

```
tar xzvf parabricks_sample.tar.gz
```

The *parabricks_sample* folder should look like this when we're done:

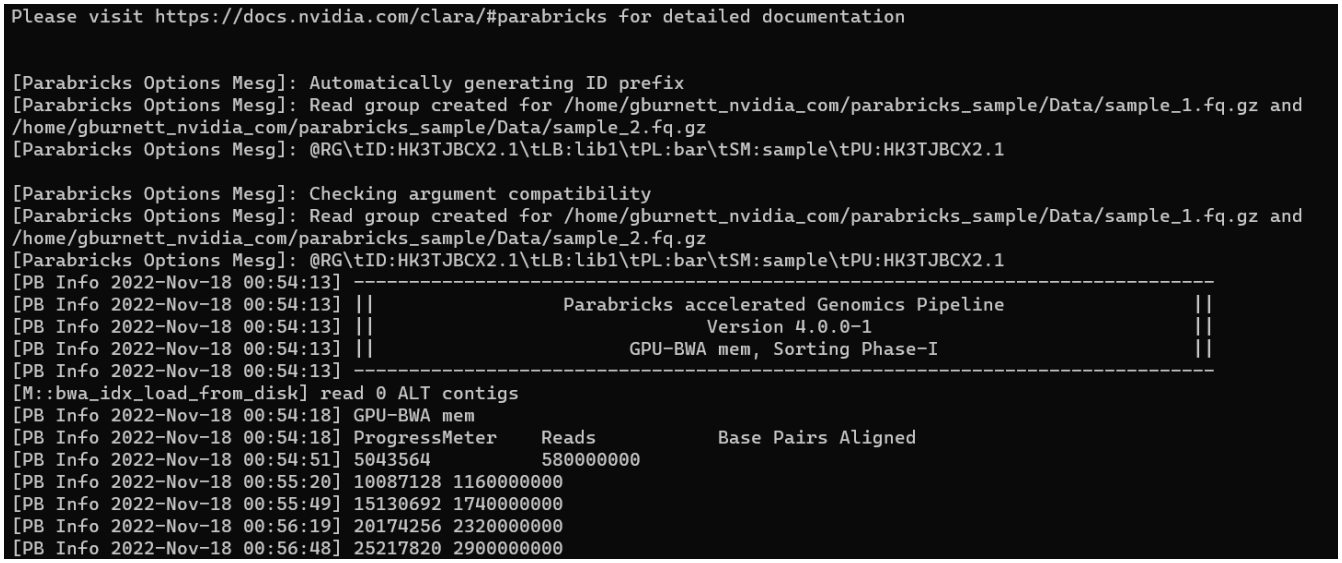


Finally, we can run any of the Parabricks pipelines on it. Let's run the germline pipeline using the following command:

```
$ docker run \ --rm \ --gpus all \ --volume `pwd`:`pwd` \ --workdir
`pwd`/parabricks_sample \ nvcr.io/nvidia/clara/clara-parabricks:4.3.1-1 \ pbrun
```

```
germline \ --ref Ref/Homo_sapiens_assembly38.fasta \ --in-fq Data/sample_1.fq.gz Data/sample_2.fq.gz \ --knownSites Ref/Homo_sapiens_assembly38.known_indels.vcf.gz.tbi \ --out-bam output.bam \ --out-variants germline.vcf \ --out-recal-file recal.txt
```

We can tell that Parabricks started correctly when we see the Parabricks banner and the ProgressMeter begins to populate with values:



This should take ~10 minutes to finish running. When it's done we should see the output files in the sample directory.

Congratulation, we've just run our first Parabricks job!

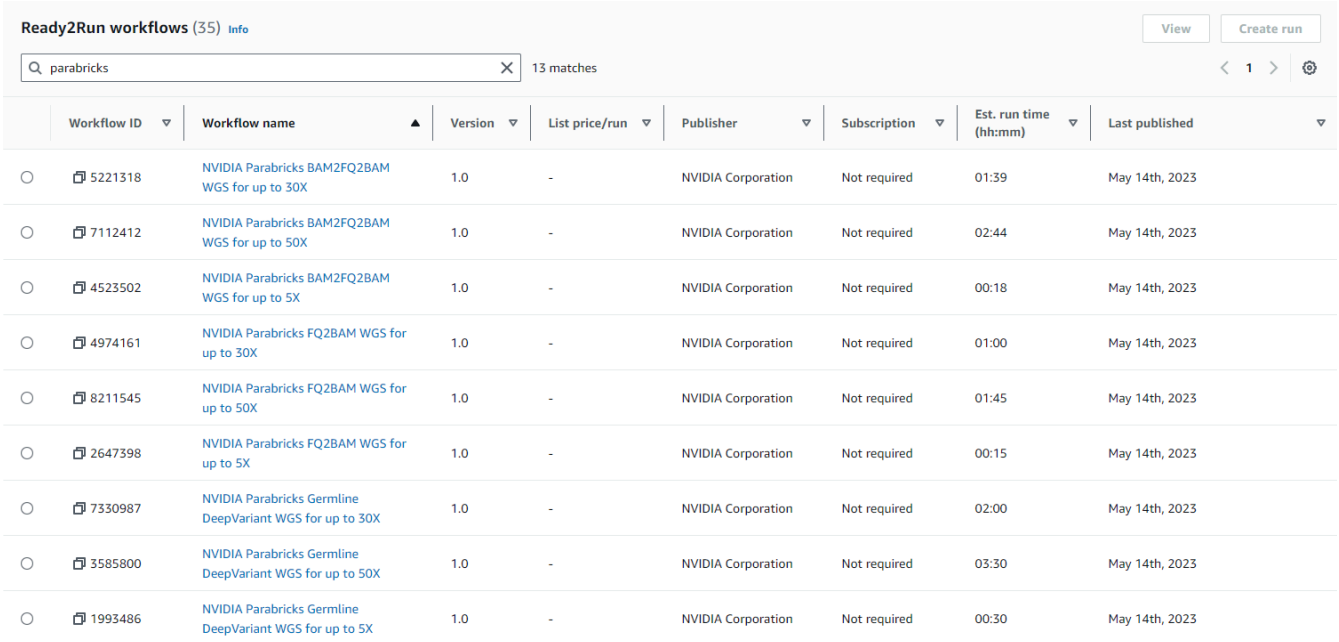
Running Workflows on Amazon HealthOmics

ReadyRun Workflows

Ready2Run Workflows are the pre-made workflows available to anyone on AWS HealthOmics.

Navigate to the AWS HealthOmics homepage and click on “Ready2Run workflows”.

In the search bar type “parabricks” to see all the available Parabricks workflows.



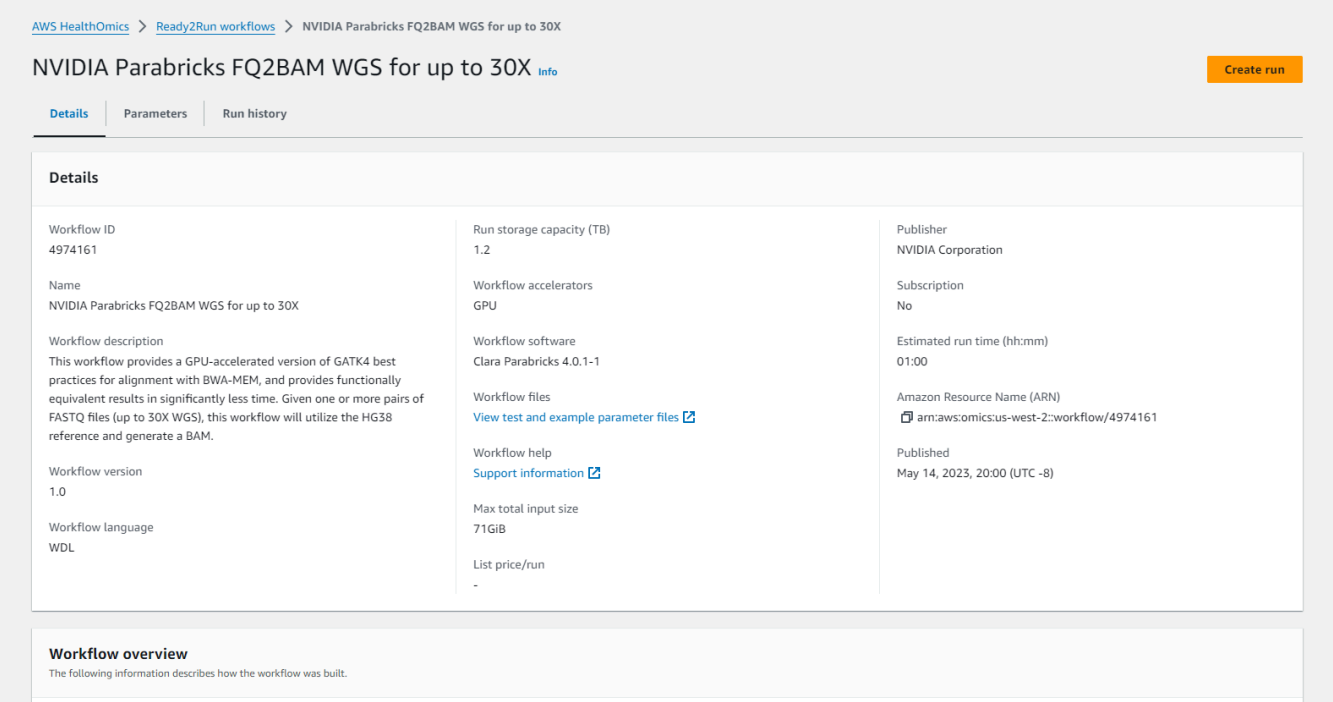
Ready2Run workflows (35) Info

View | Create run

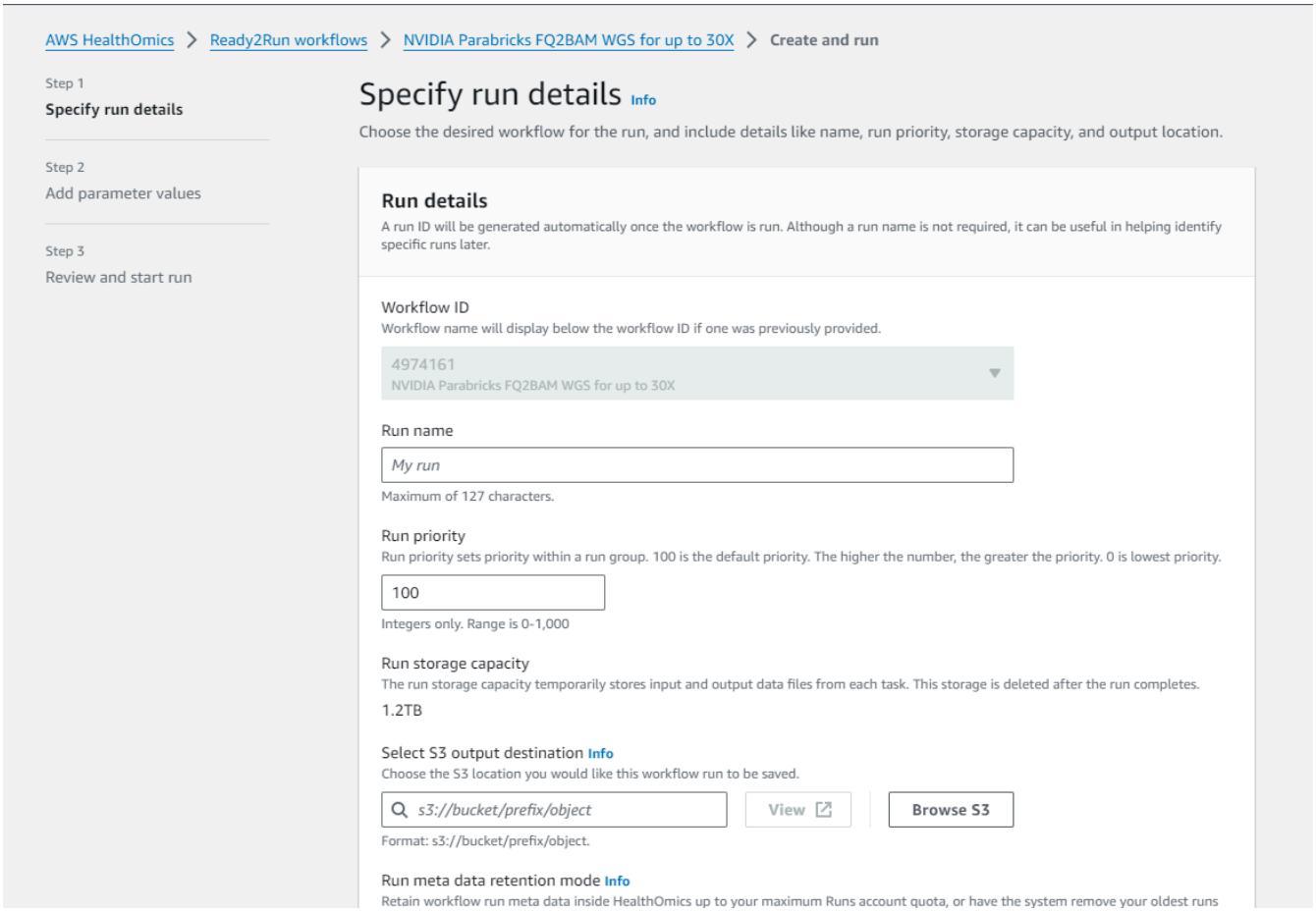
parabricks | 13 matches | 1

Workflow ID	Workflow name	Version	List price/run	Publisher	Subscription	Est. run time (hh:mm)	Last published
5221318	NVIDIA Parabricks BAM2FQ2BAM WGS for up to 30X	1.0	-	NVIDIA Corporation	Not required	01:39	May 14th, 2023
7112412	NVIDIA Parabricks BAM2FQ2BAM WGS for up to 50X	1.0	-	NVIDIA Corporation	Not required	02:44	May 14th, 2023
4523502	NVIDIA Parabricks BAM2FQ2BAM WGS for up to 5X	1.0	-	NVIDIA Corporation	Not required	00:18	May 14th, 2023
4974161	NVIDIA Parabricks FQ2BAM WGS for up to 30X	1.0	-	NVIDIA Corporation	Not required	01:00	May 14th, 2023
8211545	NVIDIA Parabricks FQ2BAM WGS for up to 50X	1.0	-	NVIDIA Corporation	Not required	01:45	May 14th, 2023
2647398	NVIDIA Parabricks FQ2BAM WGS for up to 5X	1.0	-	NVIDIA Corporation	Not required	00:15	May 14th, 2023
7330987	NVIDIA Parabricks Germline DeepVariant WGS for up to 30X	1.0	-	NVIDIA Corporation	Not required	02:00	May 14th, 2023
3585800	NVIDIA Parabricks Germline DeepVariant WGS for up to 50X	1.0	-	NVIDIA Corporation	Not required	03:30	May 14th, 2023
1993486	NVIDIA Parabricks Germline DeepVariant WGS for up to 5X	1.0	-	NVIDIA Corporation	Not required	00:30	May 14th, 2023

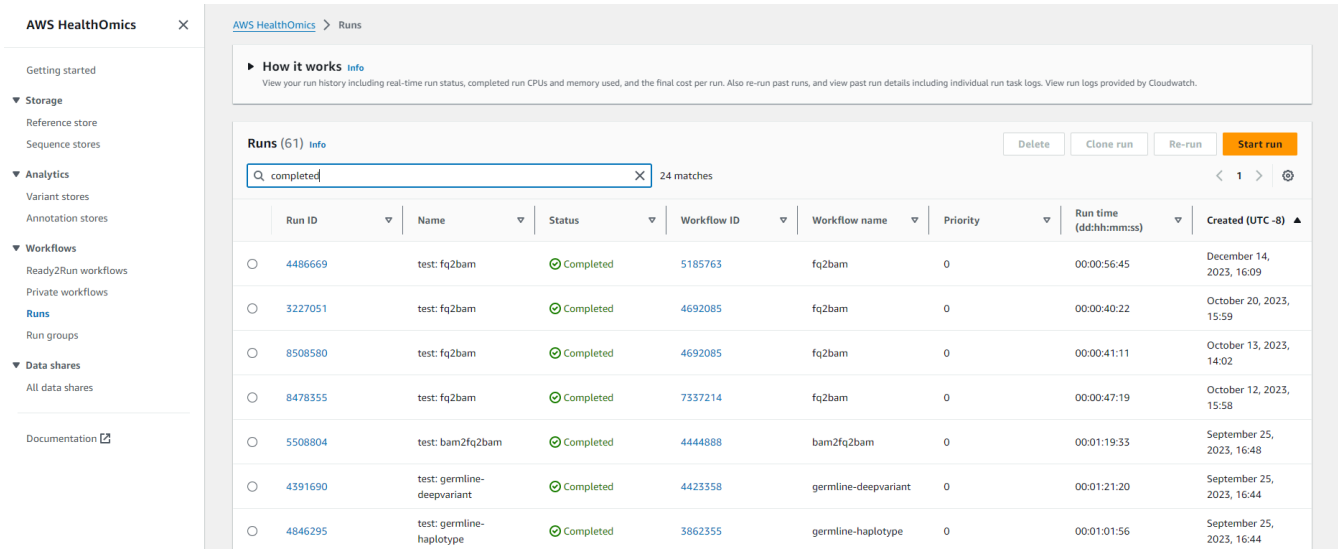
Clicking on any workflow will take us to the workflow homepage where we can see descriptions and diagrams for what the workflow does, parameters that are accepted, and the run history.



Click on “Create run” and enter a name, a destination for the output files, and the input parameters.

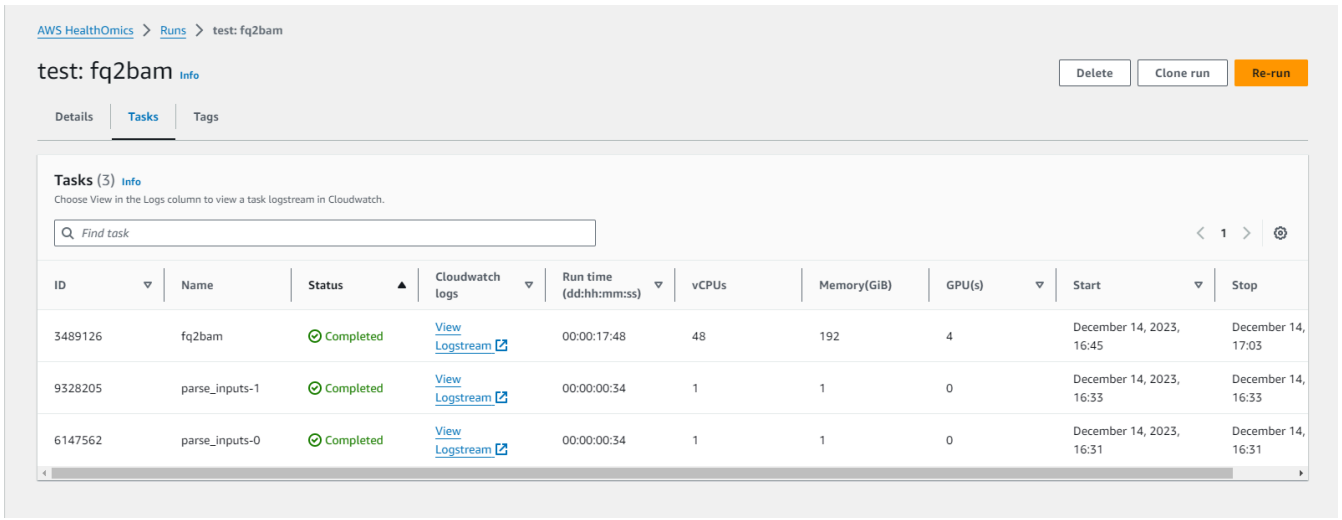


Back in the HealthOmics console, you can click on “Runs” in the left sidebar and see the job as it runs and while it’s completed:



Clicking on the job will show you information such as:

- if the job completed,
- what the inputs were,
- and where the outputs are.



And that's it! Running on AWS HealthOmics Ready2Run is designed to be simple and intuitive. You can run any number of the provided workflows using these same steps.

Private Workflows

For users who want more control over how the workflows run, we provide Private Workflows as well. These are edited locally and then run on AWS HealthOmics platform, so users can take advantage of the HealthOmics console while maintaining flexibility in the workflows themselves.

The Parabricks Private Workflows and full instructions are available in our GitHub repository.

Closing Remarks

We encourage you to expand on the demo in this guide by using your own data, trying other pipelines, and generally exploring what Parabricks has to offer. Check out the documentation for more information about the different pipelines available. You can also find our online developer community on the Parabricks forum, where you can ask questions and search through answers while you are learning how to use Parabricks.