



## **Running NVIDIA Parabricks on OCI**

# Table of contents

What is NVIDIA Parabricks?

---

Starting a GPU Instance for Parabricks Installation

---

Sign into the GPU instance to install and test Parabricks

---

Download and Install Parabricks

---

Closing Remarks

---

# List of Figures

Figure 0. ControlConsoleGUI

---

Figure 1. Marketplaces

---

Figure 2. NvidiaApps

---

Figure 3. NvidiaGpuCloud

---

Figure 4. CreateComputeInstance1

---

Figure 5. CreateComputeInstance2

---

Figure 6. CreateComputeInstance3

---

Figure 7. Icon

---

Figure 8. CreateComputeInstance4

---

Figure 9. SshCommand

---

Figure 10. WhichCommand

---

Figure 11. LsblkCommand

---

Figure 12. DockerPullCommand

---

Figure 13. TarCommand

---

Figure 14. ProgressMeter

---

Figure 15. Output

---

This guide shows how to run Parabricks on a GPU shape on OCI.

## 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.

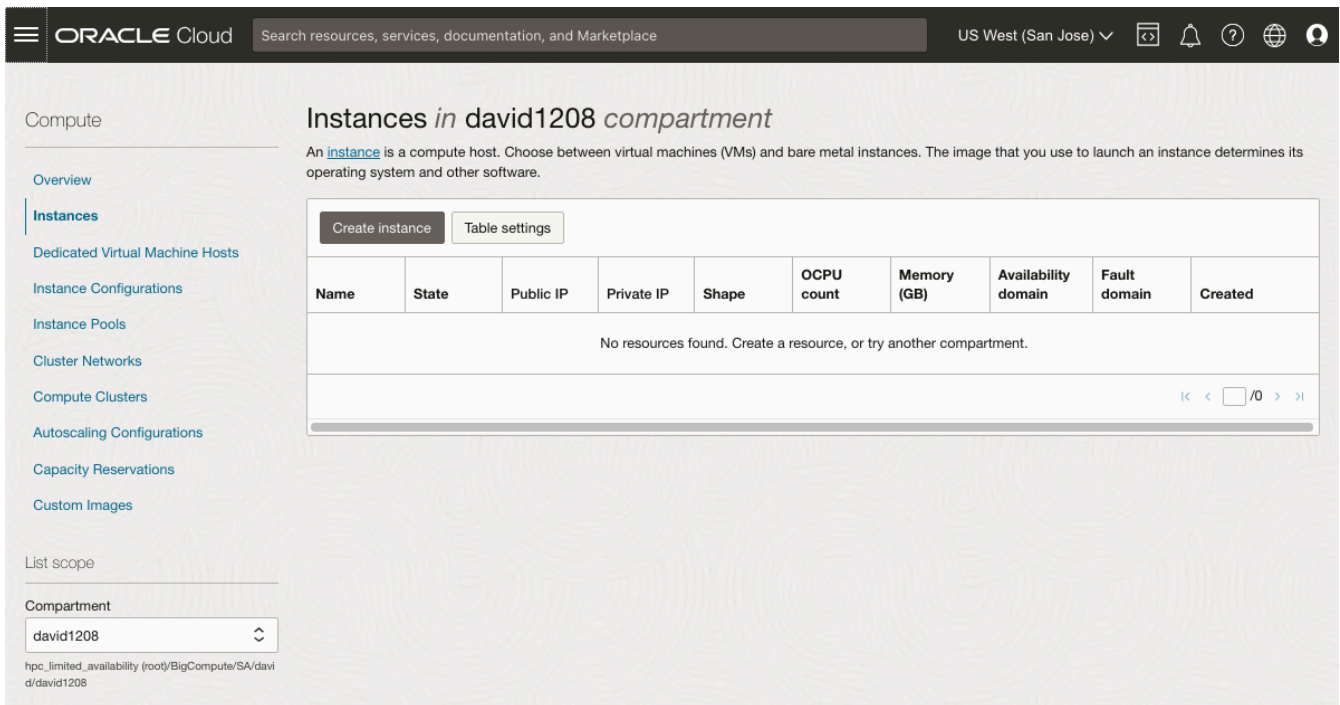
More information is available on the [Parabricks Product Page](#).

Detailed installation, usage, and tuning information is available in the [Parabricks user guide](#).

## Starting a GPU Instance for Parabricks Installation

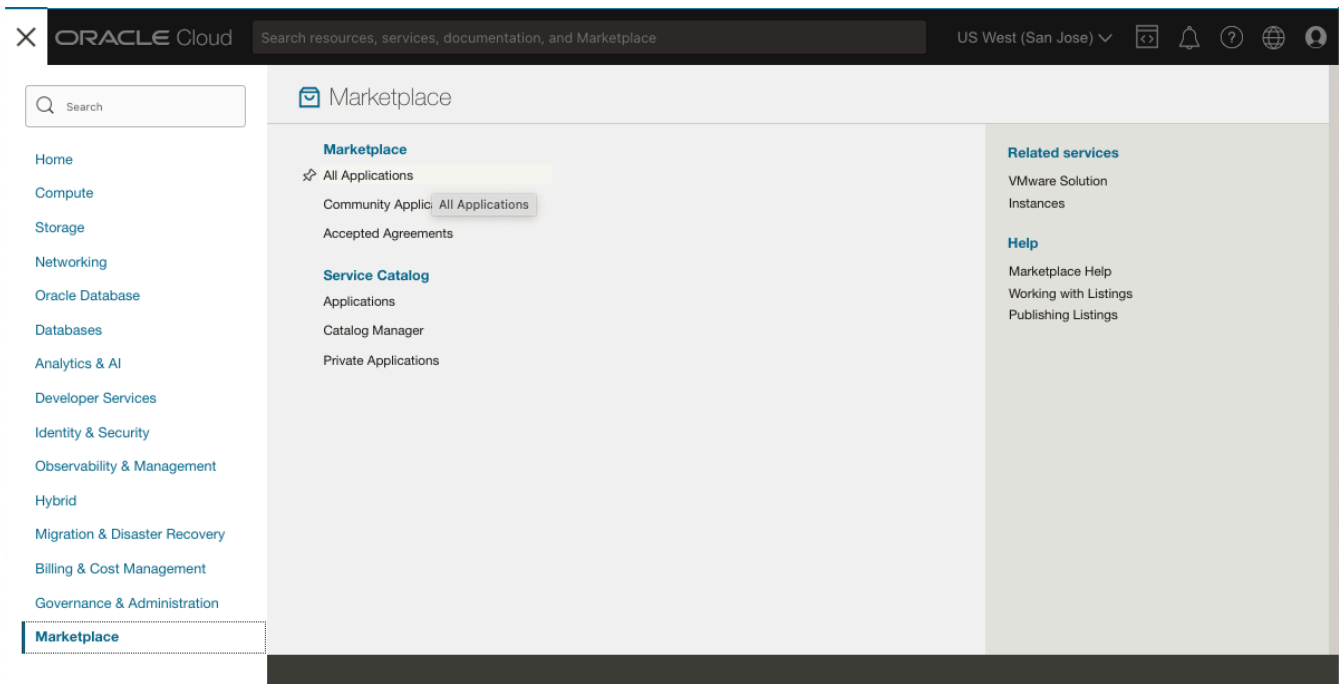
In this section, we will show how to start an EC2 instance on OCI. |

Begin by navigating to the OCI control console [GUI](#). After signing into your account, the GUI should look something like this:

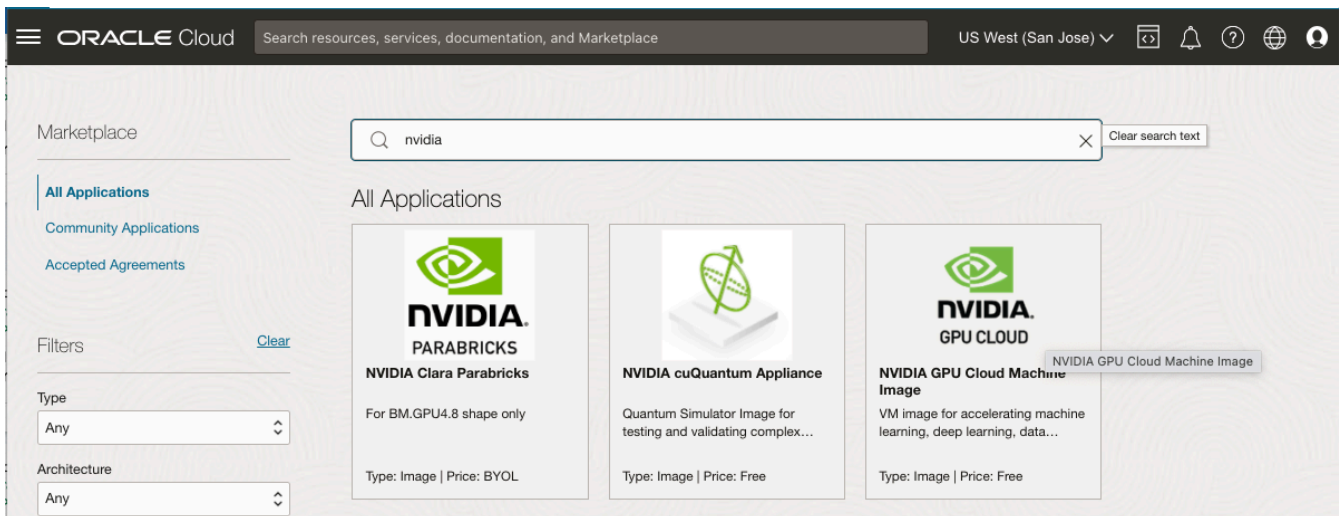


Tip: you may need to click the “3 white bars” in the top left, and then click “Instances” in the left sidebar before your GUI reaches this display. Double check your compartment in the left sidebar is correctly selected, and the cloud region in top right is also correctly selected.

Now, click on the “3 white bars” in top left, and select “Marketplaces” which is the last entry in the left sidebar. The menu page will be updated to something like this:



Click “All Applications” in the menu, and a search bar will pop up. Search for “nvidia” and you will see multiple NVIDIA application tiles popping out, as in the following example page.



Click the one marked “NVIDIA GPU CLOUD” (NGC), and you will be brought to the launch page for NGC, as follows:

The screenshot shows the Oracle Cloud Marketplace interface for the NVIDIA GPU Cloud Machine Image. The page is titled "NVIDIA GPU Cloud Machine Image" and includes a description: "VM image for accelerating machine learning, deep learning, data science, and HPC workloads." It also states: "NVIDIA GPU Cloud is a GPU-accelerated platform optimized for deep learning and scientific computing." The categories listed are "Developer Tools, Big Data".

On the right side, there is a configuration panel with the following details:

- Type: Image
- Version: 20230425 (4/25/2023...)
- Compartment: david1208
- Software price per OCPU: \$0.00/hr
- Terms of use:  I have reviewed and accept the [Oracle terms of use](#).
- Launch Instance button
- Reminder: Patch the instance once installed.

At the bottom, there are sections for "App by Oracle" and "Support". The "App by Oracle" section describes NVIDIA NGC as a hub for GPU-optimized software and mentions NVIDIA AI Enterprise. The "Support" section provides contact information and links to "NVIDIA Enterprise Support (paid)" and "Free support through NVIDIA Developer Forum".

Version Details:

- Version: 20230425
- Release Date: Apr 25, 2023, 00:00 UTC
- Ubuntu Server: 20.04.6 LTS
- NVIDIA Driver: 530.30.02
- Docker CE: 23.0.4-1
- NVIDIA Container Toolkit v1.13.1-1
- Mellanox OFED 5.9-0.5.6.0

Please review the summary in “Version Details”. Also ensure the compartment is correctly selected. Review and accept the Oracle terms of use before checking the box. Click “Launch Instance” when you are ready.

Now you will be brought to the GUI for Create compute instance, as follows:

The screenshot shows the Oracle Cloud 'Create compute instance' page. At the top, there is a navigation bar with the Oracle Cloud logo, a search bar, and the region 'US West (San Jose)'. Below the navigation bar, the page title is 'Create compute instance'. The main content area contains the following sections:

- Instructions:** 'Create an instance to deploy and run applications, or save as a reusable Terraform stack for creating an instance with Resource Manager.'
- Name:** A text input field containing 'parabricks'.
- Create in compartment:** A dropdown menu showing 'david1208'.
- Placement:** A section with a 'Collapse' link. It contains the text 'The availability domain helps determine which shapes are available.' and an 'Availability domain' dropdown menu showing 'AD 1' with the ID 'VXpT:US-SANJOSE-1-AD-1' and a checkmark. Below this is a link 'Show advanced options'.
- Security:** A section with an 'Edit' link. It shows 'Shielded instance: Disabled'.

Review and edit the default data entry in the form. Tips:

1. Update the name of the instance as desired.
2. Double check the compartment is correctly selected.
3. Make sure the availability domain has the GPU shape that you plan to use.
4. Use default setting for "Security".

Now, scroll down the GUI to continue "Create compute instance", as follows:




ORACLE Cloud Search resources, services, documentation, and Marketplace US West (San Jose)

## Create compute instance

### Image and shape Collapse


A [shape](#) is a template that determines the number of CPUs, amount of memory, and other resources allocated to an instance. The image is the operating system that runs on top of the shape.

Image



**NVIDIA GPU Cloud Machine Image**  
VM image for accelerating machine learning, deep learning, data science, and HPC workloads Return to Marketplace

Shape



**BM.GPU4.8**  
Bare metal machine, 64 core OCPU, 2048 GB memory, 50 Gbps network bandwidth Change shape

### Networking Collapse

[Networking](#) is how your instance connects to the internet and other resources in the Console. To make sure you can [connect to your instance](#), assign a public IP address to the instance.

Primary network  
 Select existing virtual cloud network  Create new virtual cloud network  Enter subnet OCID

New virtual cloud network name:  Create in compartment:

Subnet  
 An IP address from a public subnet and an [internet gateway](#) on the VCN are required to make this instance accessible from the internet.  
 Select existing subnet  Create new public subnet

New subnet name:  Create in compartment:

CIDR block:

**i** There are additional options available when you use the Networking pages in the console. To have the full range of options, [Create a VCN](#) and [Create a Subnet](#) and then select an existing VCN and subnet when you create a compute instance.

Public IPv4 address  
 Assign a public IPv4 address  Do not assign a public IPv4 address

**i** If you're not sure whether you need a public IP address, you can always assign one later.

Review and edit the default data entry in the form. Tips:

1. The OS image should be "NVIDIA GPU Cloud Machine Image", which is what you selected earlier from the Marketplace.
2. Select the GPU shape (instance). Information on how to change the default GPU shape will be provided at Appendix A at the end of this document.

3. In "Networking" section, select "Create new virtual cloud network" if this is the first time that you create an instance in this region. Otherwise, you may choose to "Select existing virtual cloud network".
4. You can similarly "Create new public subnet" or "Select existing subnet".
5. Modify "New virtual cloud network name" and "New subnet name" as desired.
6. You can use default value for the CIDR block.
7. It's easier for the beginner to select "Assign a public IPv4 address" (the default).

Now, scroll further down to finish "Create compute instance", as follows:

ORACLE Cloud Search resources, services, documentation, and Marketplace US West (San Jose) [Icons]

## Create compute instance

### Add SSH keys

Generate an [SSH key pair](#) to connect to the instance using a Secure Shell (SSH) connection, or upload a public key that you already have.

Generate a key pair for me
  Upload public key files (.pub)
  Paste public keys
  No SSH keys

SSH public keys

Drop .pub files here. [Browse](#)

ssh-1209.key.pub x

### Boot volume

A [boot volume](#) is a detachable device that contains the image used to boot the compute instance.

Specify a custom boot volume size

[Volume performance](#) varies with volume size. Default boot volume size: 46.6 GB. When you specify a custom boot volume size, service limits apply.

Boot volume size (GB)

50

Integer between 50 GB and 32,768 GB (32 TB). Must be larger than the default boot volume size for the selected image.

Boot volume performance

VPU ⓘ **Balanced**

10 10 120

Target volume performance ⓘ

IOPS: 3000 IOPS

Throughput: 24 MB/s

Balanced choice for most workloads including those that perform random I/O such as boot disks. [Learn more](#)

Actual performance depends on the attached instance's shape. Select the appropriate instance shape to optimize performance. [Learn more](#)

Encrypt this volume with a key that you manage

By default, Oracle manages the keys that encrypt this volume, but you can choose a key from a vault that you have access to if you want greater control over the key's lifecycle and how it's used. [How do I manage my own encryption keys?](#)

[Show advanced options](#)

ⓘ Live migration can't be enabled with the current instance settings

[Which settings are compatible with live migration?](#)

View incompatibilities

Create Save as stack Cancel

Review and edit the default data entry in the form. Tips:

1. Be sure to Generate, Upload or Paste your public key
2. Increase the boot volume size from 50 GB (the default) to 250 GB. Check "Specify a custom boot volume size". The "Boot volume" GUI will be expended, allowing you to modify Boot volume size (GB). Change the default value from 50 to 200.

3. The default VPU value is 10. You may raise this value to improve boot volume performance at additional cost. Click ⓘ for more information.

**ⓘ Note**

The default disk size of 50 GB is sufficient for a Parabricks installation but it is not enough to run the validation test case.

Click “Create” to launch the compute instance. You will be brought to a new GUI page, as follows:

ORACLE Cloud Search resources, services, documentation, and Marketplace US West (Phoenix)

Compute > Instances > Instance details > Work requests

## Parabricks

Start Stop Reboot **Terminate** More actions

**RUNNING**

Instance information Shielded instance Oracle Cloud Agent Notifications Tags

### General information

Availability domain: AD-2  
 Fault domain: FD-1  
 Region: phx  
 OCID: ...gwoqqa [Show](#) [Copy](#)  
 Launched: Wed, Aug 23, 2023, 03:02:17 UTC  
 Compartment: hpc\_limited\_availability (root)/BigCompute/SA/david/david1208  
 Capacity type: On-demand

### Instance details

Virtual cloud network: [VCN-ctl-OKE-0618](#)  
 Maintenance reboot: -  
 Image: [NGC-2023.04.25-01](#)  
 Launch mode: PARAVIRTUALIZED  
 Instance metadata service: Versions 1 and 2 [Edit](#) ⓘ  
 Live migration: ❌ Incompatible [View incompatibilities](#)  
 Maintenance recovery action: Restore instance

### Shape configuration

Shape: VM.GPU.A10.1  
 OCPU count: 15  
 Network bandwidth (Gbps): 24  
 Memory (GB): 240  
 Local disk: Block storage only  
 GPU count: 1 NVIDIA® A10

### Instance access

You [connect to a running Linux instance](#) using a Secure Shell (SSH) connection. You'll need the private key from the SSH key pair that was used to create the instance.  
[Usage information for this image](#)  
 Public IP address: 129.146.60.48 [Copy](#)  
 Username: ubuntu

### Primary VNIC

Public IPv4 address: 129.146.60.48  
 Private IPv4 address: 10.0.0.138  
 Network security groups: None [Edit](#) ⓘ  
 Subnet: [subnet-20230619-0030](#)  
 Private DNS record: Enable  
 Hostname: parabricks  
 Internal FQDN: parabricks... [Show](#) [Copy](#)

### Launch options

NIC attachment type: PARAVIRTUALIZED  
 Remote data volume: PARAVIRTUALIZED  
 Firmware: UEFI\_64  
 Boot volume type: PARAVIRTUALIZED  
 In-transit encryption: Disabled  
 Secure Boot: Disabled  
 Measured Boot: Disabled  
 Trusted Platform Module: Disabled  
 Confidential computing: Disabled

It will take about 1 minute for the banner in top left to turn from brown to green color, which means the instance has finished installing basic OS software, and it will need a few more minutes before the instance is ready for you to sign in.

Find and copy the public IP address from the right column of this GUI. It is 129.146.60.48 in this example.

## Sign into the GPU instance to install and test Parabricks

You can sign into the new GPU instance using ssh (be sure to properly set up your public key). For this example, the command is

```
$ ssh ubuntu@129.146.60.48
```

```
[wbchen@wbchen-mac ssh2nodes % ssh ubuntu@129.146.60.48
Welcome to Ubuntu 20.04.6 LTS (GNU/Linux 5.15.0-1040-oracle x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:    https://landscape.canonical.com
 * Support:       https://ubuntu.com/advantage

System information as of Wed Aug 23 12:40:11 UTC 2023
```

Tip: Double check that docker is installed and available in your \$PATH, and also double check the status of the GPU card:

```
$ which docker $ nvidia-smi
```

```
(base) ubuntu@parabrick:~$ which docker
/usr/bin/docker
(base) ubuntu@parabrick:~$ nvidia-smi
Wed Aug 23 03:22:32 2023
```

NVIDIA-SMI 530.30.02			Driver Version: 530.30.02			CUDA Version: 12.1		
GPU	Name	Perf	Persistence-M	Bus-Id	Disp.A	Volatile	Uncorr.	ECC
Fan	Temp		Pwr:Usage/Cap		Memory-Usage	GPU-Util	Compute	M. MIG M.
0	NVIDIA A10	P8	On	00000000:00:04.0	Off	0%	Default	0
0%	29C		16W / 150W	0MiB / 23028MiB				N/A

Processes:								
GPU	GI	CI	PID	Type	Process name	GPU Memory Usage		
	ID	ID						
No running processes found								

Tip: use lsblk to check if you have 25GB, as follows:

```
$ lsblk
```

```
((base) ubuntu@instance-20230826-1639:~$ lsblk
NAME        MAJ:MIN RM   SIZE RO TYPE MOUNTPOINT
loop0       7:0    0  55.6M  1 loop /snap/core18/2714
loop1       7:1    0  63.3M  1 loop /snap/core20/1828
loop2       7:2    0  91.9M  1 loop /snap/lxd/24061
loop3       7:3    0  52.2M  1 loop /snap/oracle-cloud-agent/50
loop4       7:4    0  49.9M  1 loop /snap/snapd/18357
loop5       7:5    0  55.7M  1 loop /snap/core18/2790
loop6       7:6    0  77.1M  1 loop /snap/oracle-cloud-agent/65
sda         8:0    0  250G   0 disk
├─sda1      8:1    0  249.9G 0 part /
├─sda14     8:14   0     4M   0 part
└─sda15     8:15   0   106M  0 part /boot/efi
```

## Download and Install Parabricks

We will install Parabricks into the instance that we just created. Visit the [Parabricks page on NGC](#) to get the Docker pull command for the latest version of Parabricks.

Briefly, the Parabricks docker image can be obtained by running the following command:

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

```
[(base) ubuntu@parabrick:~]$ docker pull nvcr.io/nvidia/clara/clara-parabricks:4.1.0-1
[4.1.0-1: Pulling from nvidia/clara/clara-parabricks
[df6635ed1257: Pull complete
[6ceabd2ff7b0: Pull complete
[cd189d71cce3: Pull complete
[b0b6463464d3: Pull complete
[0dbb8e45df45: Pull complete
[3f834e001244: Pull complete
[56cfe69b0678: Pull complete
Digest: sha256:08009fbce83e699b6acce8a8a81cc3da94d4254589de616f0e9888a41b28490d
Status: Downloaded newer image for nvcr.io/nvidia/clara/clara-parabricks:4.1.0-1
nvcr.io/nvidia/clara/clara-parabricks:4.1.0-1
```

Parabricks is now installed! Let's run some sample data to test it. Download the sample test dataset (9.9 GB) and unpack the tar file. Unpacking the tar file will require an additional 14 GB of storage:

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



```

(base) ubuntu@parabrick:~$ wget -O parabricks_sample.tar.gz \
> "https://s3.amazonaws.com/parabricks.sample/parabricks_sample.tar.gz"
--2023-08-23 03:30:08-- https://s3.amazonaws.com/parabricks.sample/parabricks_sample.tar.gz
[Resolving s3.amazonaws.com (s3.amazonaws.com)... 52.217.165.184, 54.231.233.80, 54.231.139.112, ...
Connecting to s3.amazonaws.com (s3.amazonaws.com)|52.217.165.184|:443... connected.
HTTP request sent, awaiting response... 200 OK
[Length: 9924454379 (9.2G) [application/x-tar]
Saving to: 'parabricks_sample.tar.gz'

parabricks_sample.tar.gz      100%[=====>]  9.24G  27.7MB/s   in 7m 12s

2023-08-23 03:37:21 (21.9 MB/s) - 'parabricks_sample.tar.gz' saved [9924454379/9924454379]

(base) ubuntu@parabrick:~$ ls -tl
total 9691860
drwxrwxr-x 17 ubuntu ubuntu      4096 Apr 26 02:44 miniforge3
-rw-rw-r-- 1 ubuntu ubuntu 9924454379 Feb 21 2019 parabricks_sample.tar.gz
(base) ubuntu@parabrick:~$ tar xzvf parabricks_sample.tar.gz
parabricks_sample/
parabricks_sample/Data/
parabricks_sample/Data/sample_2.fq.gz
parabricks_sample/Data/sample_1.fq.gz
parabricks_sample/Ref/
parabricks_sample/Ref/Homo_sapiens_assembly38.fasta
parabricks_sample/Ref/Homo_sapiens_assembly38.fasta.pac
parabricks_sample/Ref/Homo_sapiens_assembly38.fasta.ann
parabricks_sample/Ref/Homo_sapiens_assembly38.known_indels.vcf.gz.tbi
parabricks_sample/Ref/Homo_sapiens_assembly38.fasta.amb
parabricks_sample/Ref/Homo_sapiens_assembly38.dict
parabricks_sample/Ref/Homo_sapiens_assembly38.fasta.fai
parabricks_sample/Ref/Homo_sapiens_assembly38.known_indels.vcf.gz
parabricks_sample/Ref/Homo_sapiens_assembly38.fasta.bwt
parabricks_sample/Ref/Homo_sapiens_assembly38.fasta.sa

```

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 Progress Meter begins to populate with values:

```
(base) ubuntu@parabricks:~$ docker run --rm --gpus all \
> -v `pwd`:`pwd` \
> -w `pwd`/parabricks_sample \
> nvcr.io/nvidia/clara/clara-parabricks:4.1.0-1 \
[> 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
Please visit https://docs.nvidia.com/clara/#parabricks for detailed documentation

[Parabricks Options Msg]: Automatically generating ID prefix
[[Parabricks Options Msg]: Read group created for /home/ubuntu/parabricks_sample/Data/sample_1.fq.gz and
/home/ubuntu/parabricks_sample/Data/sample_2.fq.gz
[Parabricks Options Msg]: @RG\tID:HK3TJBCX2.1\tLB:lib1\tPL:bar\tSM:sample\tPU:HK3TJBCX2.1

[Parabricks Options Msg]: Checking argument compatibility
[Parabricks Options Msg]: Read group created for /home/ubuntu/parabricks_sample/Data/sample_1.fq.gz and
/home/ubuntu/parabricks_sample/Data/sample_2.fq.gz
[Parabricks Options Msg]: @RG\tID:HK3TJBCX2.1\tLB:lib1\tPL:bar\tSM:sample\tPU:HK3TJBCX2.1
[PB Info 2023-Aug-23 04:01:04] -----
[PB Info 2023-Aug-23 04:01:04] ||                               Parabricks accelerated Genomics Pipeline           ||
[PB Info 2023-Aug-23 04:01:04] ||                               Version 4.1.0-1                               ||
[PB Info 2023-Aug-23 04:01:04] ||                               GPU-BWA mem, Sorting Phase-I           ||
[PB Info 2023-Aug-23 04:01:04] -----
[M::bwa_idx_load_from_disk] read 0 ALT contigs
[PB Info 2023-Aug-23 04:01:08] GPU-BWA mem
[PB Info 2023-Aug-23 04:01:08] ProgressMeter      Reads          Base Pairs Aligned
[PB Info 2023-Aug-23 04:01:21] 5043564          580000000
[PB Info 2023-Aug-23 04:01:31] 10087128      1160000000
[PB Info 2023-Aug-23 04:01:41] 15130692     1740000000
[PB Info 2023-Aug-23 04:01:52] 20174256     2320000000
[PB Info 2023-Aug-23 04:02:02] 25217820     2900000000
[PB Info 2023-Aug-23 04:02:12] 30261384     3480000000
[PB Info 2023-Aug-23 04:02:23] 35304948     4060000000
[PB Info 2023-Aug-23 04:02:33] 40348512     4640000000
```

This should take up to ~10 minutes to finish the test job, depending on your selection of GPU card(s). This is the end of the output on your screen:

```
[PB Info 2023-Aug-26 21:27:40] -----
[PB Info 2023-Aug-26 21:27:40] ||           Program:           GPU-GATK4 HaplotypeCaller           ||
[PB Info 2023-Aug-26 21:27:40] ||           Version:           4.1.0-1                               ||
[PB Info 2023-Aug-26 21:27:40] ||           Start Time:        Sat Aug 26 21:22:51 2023           ||
[PB Info 2023-Aug-26 21:27:40] ||           End Time:          Sat Aug 26 21:27:40 2023           ||
[PB Info 2023-Aug-26 21:27:40] ||           Total Time:         4 minutes 49 seconds           ||
[PB Info 2023-Aug-26 21:27:40] -----
/usr/local/parabricks/binaries/bin/htvc /home/ubuntu/test/parabricks_sample/Ref/Homo_sapiens_assembly38.fasta /home/ubuntu/test/parabricks_sample/output.bam 1 -o /home/ubuntu/test/parabricks_sample/germline.vcf -nt 5 -a /home/ubuntu/test/parabricks_sample/recal.txt
```

## 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.

© Copyright 2024, Nvidia.. PDF Generated on 06/05/2024