

Using Docker in BioHPC Cloud

Jaroslav Pillardy
Bioinformatics Facility

Part 2 – hands-on exercises

May 21 2020

Docker Workflow Summary

1. Get an image as a base for your project (e.g. centos:7, or maybe one with some software already installed)
`docker pull centos:7`
2. Start container.
`docker run -d -t centos:7 /bin/bash`
3. Work with container, modify it for your needs, install software inside.
`yum install epel-release`
`yum install htop`
4. Save your container for future use, preferably as an image: commit changes in the container to a new image and save the image to a file or push it to a hub.
`docker commit jarekp__biohpc_1 htop`
`docker save -o htop_save.tar biohpc_jarekp/htop`
5. Use it: run the software as needed.

BioHPC Docker Example: install TopHat

1. Start Ubuntu container in background
2. Connect to the container and install TopHat. Check online “how to install TopHat in Ubuntu”.
3. Verify it runs
4. Save the container as image for future use
5. Use it: run the program.

BioHPC Docker Example: install TopHat

1. Start Ubuntu container in background

```
docker1 run -d -it docker.io/biohpc/ubuntudev /bin/bash
```

```
[jarekp@cbsum1c2b014 ~]$ docker1 run -d -it docker.io/biohpc/ubuntudev /bin/bash  
Unable to find image 'docker.io/biohpc/ubuntudev:latest' locally  
Trying to pull repository docker.io/biohpc/ubuntudev ...  
sha256:d03a0a1e2247895a3e57aa8cb5cdf1c6253f759b3daba78846af1d825e19cb75: Pulling from  
docker.io/biohpc/ubuntudev
```

```
87ad106e166e: Pull complete
```

```
Digest: sha256:d03a0a1e2247895a3e57aa8cb5cdf1c6253f759b3daba78846af1d825e19cb75
```

```
Status: Downloaded newer image for docker.io/biohpc/ubuntudev:latest
```

```
ee4d845bb1633c3fe907fd3a2b217f5285bd3cfa9ca24955e4720c39f4eb8e67
```

```
[jarekp@cbsum1c2b014 ~]$ docker1 ps -a
```

CONTAINER ID	IMAGE	COMMAND	CREATED	STATUS
ee4d845bb163	docker.io/biohpc/ubuntudev	"/bin/bash"	15 minutes ago	Up 15 minutes
jarekp__biohpc_1				

```
[jarekp@cbsum1c2b014 ~]$
```

BioHPC Docker Example: install TopHat

2. Connect to the container and install TopHat

```
docker1 exec -it containerid /bin/bash
```

```
apt-get update
```

```
apt-get install tophat
```

```
[jarekp@cbsum1c2b014 ~]$ docker1 exec -it ee4d845bb163 /bin/bash
root@ee4d845bb163:/workdir# apt-get update
Hit:1 http://archive.ubuntu.com/ubuntu bionic InRelease
Get:2 http://security.ubuntu.com/ubuntu bionic-security InRelease [88.7 kB]
Get:3 http://archive.ubuntu.com/ubuntu bionic-updates InRelease [88.7 kB]
Get:4 https://cloud.r-project.org/bin/linux/ubuntu bionic-cran35/ InRelease [3626 B]
[...]
Get:11 http://security.ubuntu.com/ubuntu bionic-security/multiverse amd64 Packages [8213 B]
Get:12 http://archive.ubuntu.com/ubuntu bionic-updates/multiverse amd64 Packages [12.6 kB]
Get:13 http://archive.ubuntu.com/ubuntu bionic-updates/universe amd64 Packages [1372 kB]
Get:14 http://archive.ubuntu.com/ubuntu bionic-updates/restricted amd64 Packages [59.0 kB]
Get:15 http://archive.ubuntu.com/ubuntu bionic-backports/universe amd64 Packages [7674 B]
Get:16 http://archive.ubuntu.com/ubuntu bionic-backports/main amd64 Packages [8286 B]
Fetched 4770 kB in 2s (2952 kB/s)
Reading package lists... Done
root@ee4d845bb163:/workdir#
```

BioHPC Docker Example: install TopHat

2. Connect to the container and install TopHat

```
docker1 exec -it containerid /bin/bash
```

```
apt-get update
```

```
apt-get install tophat
```

```
root@ee4d845bb163:/workdir# apt-get install tophat
```

```
Reading package lists... Done
```

```
Building dependency tree
```

```
Reading state information... Done
```

```
The following packages were automatically installed and are no longer required:
```

```
  dbus-x11 gconf2 libavahi-glib1 libbonobo2-0 libbonobo2-common libcanberra0 libgnome-2-0 libgnome2-common  
libgnomevfs2-0
```

```
  libgnomevfs2-common liborbit-2-0 libtdb1 libvorbisfile3 sound-theme-freedesktop
```

```
[...]
```

```
Processing triggers for libc-bin (2.23-0ubuntu9) ...
```

```
Setting up libboost-thread1.58.0:amd64 (1.58.0+dfsg-5ubuntu3.1) ...
```

```
Setting up libtbb2:amd64 (4.4~20151115-0ubuntu3) ...
```

```
Setting up bowtie2 (2.2.6-2) ...
```

```
Setting up tophat (2.1.0+dfsg-1build1) ...
```

```
Processing triggers for libc-bin (2.23-0ubuntu9) ...
```

```
root@ee4d845bb163:/workdir#
```

BioHPC Docker Example: install TopHat

3. Verify it runs

```
root@ee4d845bb163:/workdir# tophat -h
tophat:
TopHat maps short sequences from spliced transcripts to whole genomes.

Usage:
  tophat [options] <bowtie_index> <reads1[,reads2,...]> [reads1[,reads2,...]] \
      [quals1,[quals2,...]] [quals1[,quals2,...]]

Options:
[...]
  --rg-date           <string>      (ISO 8601 date of the sequencing run)
  --rg-platform      <string>      (Sequencing platform descriptor)

  for detailed help see http://tophat.cbcb.umd.edu/manual.html
root@ee4d845bb163:/workdir# exit
exit
[jarekp@cbsum1c2b014 ~]$
```

BioHPC Docker Example: install TopHat

4. Save the container as image for future use

```
[jarekp@cbsum1c2b014 ~]$ docker1 ps -a
CONTAINER ID   IMAGE                                COMMAND                  CREATED          STATUS          PORTS          NAMES
ee4d845bb163  docker.io/biohpc/ubuntu:dev        "/bin/bash"            28 minutes ago  Up 28 minutes  jarekp__biohpc_1

[jarekp@cbsum1c2b014 ~]$ docker1 export -o /home/jarekp/mydockerimage.tar ee4d845bb163

[jarekp@cbsum1c2b014 ~]$ ls -alh /home/jarekp/mydockerimage.tar
-rw----- 1 jarekp jarekp 1.2G May 22 18:22 /home/jarekp/mydockerimage.tar

[jarekp@cbsum1c2b014 ~]$
```

DIY: Create an image and save it to a file or your own registry/hub on Docker Hub
See slides 49-52 from Part 1 of the workshop

BioHPC Docker Example: install TopHat

5. Use it: run the program

```
[jarekp@cbsum1c2b014 ~]$ cd /workdir/jarekp
[jarekp@cbsum1c2b014 jarekp]$ cp /shared_data/RNAseq/exercise1/ERR458494.fastq.gz .
[jarekp@cbsum1c2b014 jarekp]$ cp /shared_data/RNAseq/exercise1/R64.fa .
[jarekp@cbsum1c2b014 jarekp]$ cp /shared_data/RNAseq/exercise1/R64.gtf .
```

```
root@ee4d845bb163:/workdir# bowtie2-build R64.fa R64
```

```
Settings:
```

```
Output files: "R64.*.bt2"
```

```
Line rate: 6 (line is 64 bytes)
```

```
Lines per side: 1 (side is 64 bytes)
```

```
Offset rate: 4 (one in 16)
```

```
[...]
```

```
reverse: 1
```

```
Total time for backward call to driver() for mirror index: 00:00:07
```

```
root@ee4d845bb163:/workdir#
```

BioHPC Docker Example: install TopHat

5. Use it: run the program

```
root@ee4d845bb163:/workdir# tophat -G R64.gtf --no-novel-juncs --num-threads 2 R64 ERR458494.fastq.gz
```

```
root@ee4d845bb163:/workdir# ls -al
```

```
total 102740
drwxr-xr-x 3 4965 root      189 May 18 17:30 .
drwxr-xr-x 1 root root       51 May 18 17:23 ..
-rw-r--r-- 1 4965 4963 58566854 May 18 17:18 ERR458494.fastq.gz
-rw-r--r-- 1 root root  8248091 May 18 17:29 R64.1.bt2
-rw-r--r-- 1 root root  3039284 May 18 17:29 R64.2.bt2
-rw-r--r-- 1 root root    161 May 18 17:28 R64.3.bt2
-rw-r--r-- 1 root root  3039277 May 18 17:28 R64.4.bt2
-rw-r--r-- 1 4965 4963 12360704 May 18 17:18 R64.fa
-rw-r--r-- 1 4965 4963  8639033 May 18 17:18 R64.gtf
-rw-r--r-- 1 root root  8248091 May 18 17:29 R64.rev.1.bt2
-rw-r--r-- 1 root root  3039284 May 18 17:29 R64.rev.2.bt2
drwxr-xr-x 3 root root      175 May 18 17:33 tophat_out
```

Files are owned by root, you need to use

`docker1 claim`

command to gain ownership

Exercises

1. Run `biohpc/cowsay` in three ways (single command, interactive, in background) following examples on slides 18-22 from Part 1.
2. Stop background container (slide 26 Part 1).
3. Run container with `/workdir/userid/testdata` mapped to `/data` (slides 31-33 Part 1)
4. Replicate `htop` installation (slides 36-52 Part 1)
5. Install `tophat` in Docker and test it.