2. Common commands for docker image containers

The operating environment and software and hardware reference configuration are as follows:

- Reference model: ROSMASTER X3
- Robot hardware configuration: Arm series main control, Silan A1 lidar, AstraPro Plus depth camera
- Robot system: Ubuntu (no version required) + docker (version 20.10.21 and above)
- PC virtual machine: Ubuntu (18.04) + ROS (Melodic)
- Usage scenario: Use on a relatively clean 2D plane

2.1. Do not use sudo command

Normally, to operate docker commands, you need to prefix sudo, as follows:

```
sudo docker version
```

But after adding the docker user group, you don't need to add the sudo prefix. How to add the docker user group (run the command in the host running docker):

```
sudo groupadd docker # Add docker user group
sudo gpasswd -a $USER docker # Add the current user to the docker user group,
where $USER can automatically resolve to the currently logged in user
newgrp docker # Update docker user group
```

After adding the above command, use the [docker images] command to test. If no error is reported, it means that the sudo command is no longer needed. If the following error is reported:

```
pi@ubuntu:~$ docker images
WARNING: Error loading config file: /home/pi/.docker/config.json: open
/home/pi/.docker/config.json: permission denied
```

Then execute the following command on the host machine to solve the problem:

```
sudo chown "$USER":"$USER" /home/"$USER"/.docker -R
sudo chmod g+rwx "/home/$USER/.docker" -R
```

2.2. Help command

```
docker info # Display Docker system information, including the number of images
and containers. .
docker --help # help
```

2.3. Mirror command

1. docker pull download image

```
# Download image
jetson@ubuntu:~$ docker pull ubuntu
Using default tag: latest # Do not write tag, the default is latest
latest: Pulling from library/ubuntu
cd741b12a7ea: Pull complete # Layered download
Digest: sha256:67211c14fa74f070d27cc59d69a7fa9aeff8e28ea118ef3babc295a0428a6d21
Status: Downloaded newer image for ubuntu:latest
docker.io/library/ubuntu:latest #Real location
```

2. docker images list images

```
# List the images on the local host
jetson@ubuntu:~$ docker images
REPOSITORY TAG IMAGE ID CREATED SIZE
yahboomtechnology/ros-melodic 1.4.1 f8c914ba3cff 7 weeks ago 23.1GB
hello-world latest 46331d942d63 13 months ago 9.14kB

# explain
REPOSITORY mirrored warehouse source
TAG The tag of the image
IMAGE ID ID of the image
CREATED Image creation time
SIZE image size

# The same warehouse source can have multiple TAGs, representing different
versions of the warehouse source. We use REPOSITORY: TAG to define different
images. If you do not define the tag version of the image, docker will use the
lastest image by default!

#optional
-a: List all local images
-q: only display image id
--digests: Display summary information of the image
```

3. docker search search image

```
# Search for images
jetson@ubuntu:~$ docker search ros2
NAME DESCRIPTION STARS OFFICIAL AUTOMATED
osrf/ros2 **Experimental** Docker Images for ROS2 deve… 60 [OK]
tiryoh/ros2-desktop-vnc A Docker image to provide HTML5 VNC interfac… 11
althack/ros2 An assortment of development containers for … 7
tiryoh/ros2 unofficial ROS2 image 6
athackst/ros2 [Deprecated-> use althack/ros2] 5
uobflightlabstarling/starling-mavros2 ROS2 version of MAVROS 2
theosakamg7/ros2_java_docker Image base 1 [OK]

# docker search The name of a certain image corresponds to the image in the
DockerHub warehouse
#optional
--filter=stars=50: List the images whose collection number is not less than the
specified value.
```

4. docker rmi deletes the image

```
# Delete image
docker rmi -f image id # Delete a single
docker rmi -f Image name: tag Image name: tag # Delete multiple
docker rmi -f $(docker images -qa) # Delete all
```

2.4. Container command

A container can only be created if you have a mirror. We use the ubuntu mirror to test here. Download the mirror:

```
docker pull ubuntu
```

1. docker run runs the image to start the container

```
# Order
docker run [OPTIONS] IMAGE [COMMAND][ARG...]
# Common parameter descriptions
--name="Name" # Specify a name for the container
-d # Run the container in background mode and return the container id!
-i # Run the container in interactive mode, used with -t
-t #Reassign a terminal to the container, usually used with -i
-P # Random port mapping (uppercase)
-p #Specify port mapping (summary), generally there are four ways to write it
ip:hostPort:containerPort
ip::containerPort
```

```
hostPort:containerPort (commonly used)
containerPort

#test
jetson@ubuntu:~$ docker images
REPOSITORY TAG IMAGE ID CREATED SIZE
yahboomtechnology/ros-melodic 1.4.1 f8c914ba3cff 7 weeks ago 23.1GB
ubuntu latest bab8ce5c00ca 6 weeks ago 69.2MB
hello-world latest 46331d942d63 13 months ago 9.14kB

#Use ubuntu to start the container in interactive mode and execute the /bin/bash
command in the container!
jetson@ubuntu:~$ docker run -it ubuntu:latest /bin/bash
root@c54bf9efae47:/# ls
bin boot dev etc home lib media mnt opt proc root run sbin srv sys tmp usr var
root@c54bf9efae47:/# exit # Use exit to exit the container and return to the host
exit
jetson@ubuntu:~$
```

2. docker ps lists all running containers

```
# Order
docker ps [OPTIONS]
# Common parameter descriptions
-a # List all currently running containers + historically run containers
-l # Display recently created containers
-n=? # Display the last n created containers
-q # Silent mode, only the container number is displayed.

#test
jetson@ubuntu:~$ docker ps -a
CONTAINER ID IMAGE COMMAND CREATED STATUS PORTS NAMES
c54bf9efae47 ubuntu:latest "/bin/bash" 2 hours ago Up 4 seconds funny_hugle
3b9c01839579 hello-world "/hello" 3 hours ago Exited (0) 3 hours ago jovial_brown
```

3. Exit the container

```
exit # Container stops exiting
ctrl+P+Q # Container exits without stopping
```

4. Enter the running container from multiple terminals

```
# Command 1
docker exec -it container id bashShell

# test
jetson@ubuntu:~$ docker ps -a
CONTAINER ID IMAGE COMMAND CREATED STATUS PORTS NAMES
c54bf9efae47 ubuntu:latest "/bin/bash" 2 hours ago Up 4 seconds funny_hugle
```

```
3b9c01839579 hello-world "/hello" 3 hours ago Exited (0) 3 hours ago jovial_brown
jetson@ubuntu:~$ docker exec -it c5 /bin/bash # The container ID can be
abbreviated, as long as it can uniquely identify the container.
root@c54bf9efae47:/#

# Command 2
docker attach container id

# test
jetson@ubuntu:~$ docker ps -a
CONTAINER ID IMAGE COMMAND CREATED STATUS PORTS NAMES
c54bf9efae47 ubuntu:latest "/bin/bash" 2 hours ago Up 35 seconds funny_hugle
3b9c01839579 hello-world "/hello" 3 hours ago Exited (0) 3 hours ago jovial_brown
jetson@ubuntu:~$ docker attach c5 # The ID of the container can be abbreviated,
as long as it can uniquely identify the container.
root@c54bf9efae47:/#

# the difference
# exec opens a new terminal in the container and can start a new process
# attach directly enters the terminal of the container startup command and will
not start a new process.
```

5. Start and stop the container

```
docker start (container id or container name) # Start the container
docker restart (container id or container name) # Restart the container
docker stop (container id or container name) # Stop the container
docker kill (container id or container name) #Force stop the container
```

6. Delete the container

```
docker rm container id # Delete the specified container
docker rm -f $(docker ps -a -q) # Delete all containers
docker ps -a -q|xargs docker rm # Delete all containers
```

2.5. Other commonly used commands

1. View the process information running in the container and support ps command parameters.

```
# Order
docker top container id

# test
jetson@ubuntu:~$ docker ps -a
CONTAINER ID IMAGE COMMAND CREATED STATUS PORTS NAMES
c54bf9efae47 ubuntu:latest "/bin/bash" 2 hours ago Up 2 minutes funny_hugle
3b9c01839579 hello-world "/hello" 3 hours ago Exited (0) 3 hours ago jovial_brown
jetson@ubuntu:~$ docker top c5
UID PID PPID C STIME TTY TIME CMD
root 9667 9647 0 14:20 pts/0 00:00:00 /bin/bash
```

2. View the metadata of the container/image

```
# Order
docker inspect container id

#Test to view container metadata
jetson@ubuntu:~$ docker ps -a
CONTAINER ID IMAGE COMMAND CREATED STATUS PORTS NAMES
c54bf9efae47 ubuntu:latest "/bin/bash" 2 hours ago Up 4 minutes funny_hugle
3b9c01839579 hello-world "/hello" 3 hours ago Exited (0) 3 hours ago jovial_brown
jetson@ubuntu:~$ docker inspect c54bf9efae47
[
     {
     # The complete id, the container id above is the first few digits of the
intercepted id.
         "Id":
"c54bf9efae471071391202a8718b346d9af76cb1ff17741e206280603d6f0056",
         "Created": "2023-04-24T04:19:46.232822024Z",
         "Path": "/bin/bash",
         "Args": [],
         "State": {
             "Status": "running",
             "Running": true,
             "Paused": false,
             "Restarting": false,
             "OOMKilled": false,
             "Dead": false,
             "Pid": 9667,
             "ExitCode": 0,
             "Error": "",
             "StartedAt": "2023-04-24T06:20:58.508213216Z",
             "FinishedAt": "2023-04-24T06:19:45.096483592Z"
         },
. . . .


#Test to view image metadata
jetson@ubuntu:~$ docker images
REPOSITORY TAG IMAGE ID CREATED SIZE
ubuntu latest bab8ce5c00ca 6 weeks ago 69.2MB
hello-world latest 46331d942d63 13 months ago 9.14kB
```

```
jetson@ubuntu:~$ docker inspect bab8ce5c00ca
[
     {
          "Id":
"sha256:bab8ce5c00ca3ef91e0d3eb4c6e6d6ec7cffa9574c447fd8d54a8d96e7c1c80e",
          "RepoTags": [
               "ubuntu:latest"
          ],
          "RepoDigests": [

"ubuntu@sha256:67211c14fa74f070d27cc59d69a7fa9aeff8e28ea118ef3babc295a0428a6d21"
          ],
          "Parent": "",
          "Comment": "",
          "Created": "2023-03-08T04:32:41.063980445Z",
          "Container":
"094fd0c521be8c84d81524e4a5e814e88a2839899c56f654484d32d171c7195b",
          "ContainerConfig": {
               "Hostname": "094fd0c521be",
               ..............
               "Labels": {
                    "org.opencontainers.image.ref.name": "ubuntu",
                    "org.opencontainers.image.version": "22.04"
               }
          },
          "DockerVersion": "20.10.12",
          "Author": "",
          "Config": {
               "Hostname": "",
               .........
               "Labels": {
                    "org.opencontainers.image.ref.name": "ubuntu",
                    "org.opencontainers.image.version": "22.04"
               }
          },
          "Architecture": "arm64",
          "Variant": "v8",
          "Os": "linux",
          "Size": 69212233,
          "VirtualSize": 69212233,
          "GraphDriver": {
               "Data": {
                    "MergedDir":
"/var/lib/docker/overlay2/8418b919a02d38a64ab86060969b37b435977e9bbdeb6b0840d4eb6
98280e796/merged",
                    "UpperDir":
"/var/lib/docker/overlay2/8418b919a02d38a64ab86060969b37b435977e9bbdeb6b0840d4eb6
98280e796/diff",
                    "WorkDir":
"/var/lib/docker/overlay2/8418b919a02d38a64ab86060969b37b435977e9bbdeb6b0840d4eb6
98280e796/work"
               },
               "Name": "overlay2"
          },
          "RootFS": {
               "Type": "layers",
```

```
            "Layers": [
                "sha256:874b048c963ab55b06939c39d59303fb975d323822a4ea48a02ac8dc635ea371"
            ]
        },
        "Metadata": {
            "LastTagTime": "0001-01-01T00:00:00Z"
        }
    }
]
```

2.6. Command summary