1. In windows on the company's laptop, gain admin, download docker desktop and install it, follow the installation instruction and add your current user account to the docker user group, reboot the laptop a few times will do.

2. Download some container image using the pull request instruction that should be accompanied. The image will automatically show up in the docker desktop.

3. Run the image but with setting the shared folder in the dialog box.

4. Run the command line tool (cli). First execute "/bin/bash" and this will allow you to activate the left and right arrow on the keyboard.

Install docker on Linux cluster:

1. Installation for the first time:

*sudo docker pull rocker/r-devel-ubsan-clang*

2. Run container bash and mount host directory to /mnt:

*sudo docker run -v /finance\_develop/Charlie:/mnt -it rocker/r-devel-ubsan-clang bash*

3. After installing some softwares in the container and you want to save it:

3.0 Exit docker container.

3.1 Show the image ID: *sudo docker ps -a*

3.2 Commit the change: *sudo docker commit afd56440470c rocker/rdebugcpp*

3.3 Save the container image somewhere:

*sudo docker save -o dockerImage/rdebugcpp.tar rocker/rdebugcpp*

3.4 In another Linux development instance, load the image:

*sudo docker load -i /finance\_develop/Charlie/dockerImage/rdebugcpp.tar*

and then run the container. Sometimes the command cannot find the file. This might be related to the instability of the distributed computing environment. Fixed this via first run *ls dockerImage*, then the docker command will work.

3.5: *sudo docker run -v $pwd:/mnt -it rocker/rdebugcpp bash; cd /mnt; Rdevel;*

So together:

***sudo docker load -i /finance\_develop/Charlie/dockerImage/rdebugcpp.tar***

***sudo docker run -v $(pwd):/mnt -it rocker/rdebugcpp bash***

***cd /mnt***

***Rdevel***

And this is written into a script in /home/.local/bin/debugRcpp.sh:

***sudo docker load -i /finance\_develop/Charlie/dockerImage/rdebugcpp.tar***

***sudo docker run -v $(pwd):/mnt -it rocker/rdebugcpp bash -c "cd /mnt && Rdevel"***

Notice that the way of executing multiple commands in docker at start.

So in the future, all you need is *debugRcpp.sh* to enter the R debug environment.