

## 在Docker下安装NetDEM和CFDDem。

提前下载准备 (<https://www.jianguoyun.com/p/DfPZbg8Qo6K6CBiqn90FIAA> 有效期: 10.31 23:59)

1. docker (Docker Desktop Installer.exe)
2. VSCode (VSCodeSetup-x64-1.94.2.exe)
3. Ubuntu22.04 (ubuntu\_image.tar)
4. paraview (ParaView-5.13.1-Windows-Python3.10-msvc2017-AMD64.msi)
5. netdem (netdem-1.3-cp310-cp310-linux\_x86\_64.whl) 、 openfoam (openfoam\_customized.zip) 、 cfdem (98-cfdem\_solver.zip)
6. example netdem (netdem.zip) 、 cfdem (99-cfdem\_example.zip)

### 安装docker & VSCode:

默认选项安装 (一路确认) , 重启电脑。

注册&登录账号 (也可以直接skip跳过) 安装完成后进入右上角

设置->resource中更改文件夹路径至剩余空间较大的硬盘 (默认为C盘)

### 导入ubuntu的镜像,运行这个镜像.

```
docker load -i ubuntu_image.tar
docker run -it
sha256:97271d29cb7956f0908cfb1449610a2cd9cb46b004ac8af25f0255663eb364ba
```

### 安装VSCode的docker扩展

打开VSCode, 在左侧栏中打开扩展, 搜索docker。下载docker插件。

### 在VSCode中打开docker中的container。

1. 点击左下角蓝色><图标 (Open a remote window) 。
2. 弹出的列表 单击 `dev container` , 等待安装后, 会有文件夹弹窗, 关闭弹窗。
3. 点击左下角蓝色><图标, `attach to running container` , 点击并选择 `sha256:97271d29cb7956f0908cfb1449610a2cd9cb46b004ac8af25f0255663eb364ba` , 会出现新的VSCode窗口, 成功, 默认为root权限,

以下命令如非注明在win10中使用, 则均在VSCode终端使用

创建有管理员权限的普通用户: (cfdem不支持直接在root用户中运行, 因此需要以普通用户的身份运行)

新建终端, 选择上方的 `terminal` , 选择 `new terminal`  
在终端输入以下命令:

安装sudo:

```
apt update  
apt install sudo unzip
```

添加用户：**student**为用户名，设置密码，其他选项空着

```
adduser student
```

添加用户**sudo**权限：

```
usermod -aG sudo student
```

从**root**切换到**student**：

```
su - student
```

## 安装python3.10:

安装所需依赖：

```
sudo apt install -y build-essential libssl-dev zlib1g-dev libffi-dev libsqlite3-dev
```

安装python3.10:

```
sudo apt install python3.10
```

验证是否安装成功：(终端会显示正确版本)

```
python3.10 --version
```

安装pip:

```
sudo apt-get install python3-pip
```

## 安装netdem:

从主机上把安装包拷贝到容器中（在win10命令行操作）

在win+R中输入cmd 新建另一个终端。

```
docker cp 目录\文件名 容器id: 目录
```

容器id在docker的container界面中可以找到并复制.(注意：容器的id是随机的！)

示例：（注意windows中斜杠和Linux中是反着的）

```
docker cp C:\Users\Administrator\Desktop\netdem-1.3-cp310-cp310-linux_x86_64.whl
ed45a5e9f2087e2b9f8212ed9367096334e6db894ff53aba3726569632b9f9cd:/home/student/netdem-
1.3-cp310-cp310-linux_x86_64.whl
docker cp C:\Users\Administrator\Desktop\netdem.zip
ed45a5e9f2087e2b9f8212ed9367096334e6db894ff53aba3726569632b9f9cd:/home/student/netdem.zip
```

### 安装libopenmpi-dev:

```
sudo apt-get install libopenmpi-dev
```

### 安装netdem:

```
pip install /home/student/netdem-1.3-cp310-cp310-linux_x86_64.whl
```

### 测试

终端输入python3进入python环境（ctrl+D退出），输入 `import netdem`，没报错则安装成功

### 安装openfoam:

#### 安装git、rsync、flex、vim

```
sudo apt-get install git rsync flex vim
```

#### 从apaam安装openfoam并编译:

```
sudo git clone https://github.com/apaam/openfoam_customized.git /安装目录
```

#### 或者

下载压缩包，从windows中复制入docker中。

```
docker cp C:\Users\Administrator\Desktop\openfoam_customized.zip
ed45a5e9f2087e2b9f8212ed9367096334e6db894ff53aba3726569632b9f9cd:/home/student/openfoam_c
ustomized.zip
```

### 进行安装

```
cd /安装目录
sudo make
vim ~/.bashrc #打开.bashrc配置环境变量（注：普通用户和root用户的.bashrc不同）
```

按i进入编辑模式，在末尾加上以下几行（路径自行调整）：

```
export path_openfoam=/home/student/openfoam_customized/OpenFOAM-build
alias openfoam_init='source $path_openfoam/etc/bashrc'
openfoam_init
echo "using openfoam=$path_openfoam"
```

输入 :wq 保存并退出 ~/.bashrc 应用更改：

```
source ~/.bashrc
```

检验，输入：

```
which blockMesh
```

出现openfoam的路径则配置成功

## 安装cfddem：

```
docker cp C:\Users\Administrator\Desktop\cfddem-0.1.1-Linux.tar.gz  
ed45a5e9f2087e2b9f8212ed9367096334e6db894ff53aba3726569632b9f9cd:/home/student/cfddem-  
0.1.1-Linux.tar.gz
```

解压：

```
sudo tar -xvf cfddem-0.1.1-Linux.tar.gz -C /目标路径  
vim ~/.bashrc #打开.bashrc配置环境变量（注：普通用户和root用户的.bashrc不同）
```

按i进入编辑模式，在末尾加上以下几行（路径自行调整）：

同样的，打开.bashrc配置环境变量：

```
export path_cfddem=/home/student/cfddem-0.1.1-Linux/CFDDEM  
alias cfddem_init='source $path_cfddem/etc/bashrc'  
cfddem_init  
echo "using cfddem=$path_cfddem"
```

输入 :wq 保存并退出 ~/.bashrc 应用更改：

```
source ~/.bashrc
```

检验，输入：

```
which interIBdem
```

出现cfddem的路径则配置成功

## 给算例文件赋予权限：

拷贝到ubuntu的算例文件夹需要赋予全部的读写权限才能正常运行：

```
sudo chmod -R 777 文件夹名
```

## 拉取ubuntu22.04镜像：

win+r 输入cmd，命令行输入

```
docker pull ubuntu:22.04
```

成功后继续输入：

```
docker run -it ubuntu:22.04
```

可以看见root开头字样，说明加载成功，在docker界面可以看到加载的ubuntu22.04容器，输入exit退出，在docker desktop中启动和停止ubuntu22.04容器