WRF-HYDRO

Overview WRF-Hydro Docker Training Container

This container is used for WRF-Hydro training sessions and can be run locally.

This container includes the following:

- Ubuntu base image
- All system libraries required by WRF-Hydro
- Text editors - VIM, Nano, Emacs
- git version control system

##Requirements

The easiest and recommended way to run the training lessons is via the wrfhydro/training Docker container, which has all software dependencies and data pre-installed.

- Docker >= v.17.12
- Web browser (Google Chrome recommended)

Where to get help and/or post issues

If you have general questions about Docker, there are ample online resources including the excellent Docker documentation at https://docs.docker.com/.

If you have questions regarding the lessons please contact us here https://ral.ucar.edu/projects/wrf_hydro/contact.

The best place to ask questions or post issues with these lessons is via the Issues page of the GitHub repository at https://github.com/NCAR/wrf_hydro_training/issues.

How to run

Make sure you have Docker installed and that it can access your localhost ports. Most out-of-the-box Docker installations accepting all defaults will have this configuration.

NOTE: THE DEFAULT DOCKER CONFIGURATION IS FOR 2 CPUS, YOU MUST HAVE AT LEAST 2 CPUS AVAILABLE TO THE DOCKER DAEMON FOR THIS TRAINING

Step 1: Open a terminal or PowerShell session

Step 2: Pull the wrfhydro/training Docker container for the desired code version Each training container is specific to a release version of the WRF-Hydro source code, which can be found at https://github.com/NCAR/wrf_hydro_nwm_public/releases. Issue the following command in your terminal to pull a specific version of the training corresponding to your code release version. In this example, we will pull the training container for v5.0.x.

```
docker pull wrfhydro/costarica-training
```

Step 3: Start the training Docker container Issue the following command in your terminal session to start the training Docker container.

```
docker run --name wrf-hydro-training -p 8888:8888 -it wrfhydro/costarica-training
```

Note: Port forwarding is setup with the -p 8888:8888 argument, which maps your localhost port to the container port. If you already have something running on port 8888 on your localhost you will need to change the first number to an unused port on your system

The container will start and perform a number of actions before starting the training.

- First, the container will pull the model code corresponding to the specified major version, in this case v5.1.x *(v5.1.1 currently)*
- Second, the container will pull an example test case compatible with the model code release.
- Third, the container will pull the training lessons corresponding to the specified major version.
- Fourth, the container will launch a Jupyter Notebook server and echo the address to your terminal.

Step 4: Open the Jupyter Notebook lessons All lessons for this training are contained in the `/home/docker/wrf-hydro-training/lessons` folder. The lessons are interactive and can execute code commands live. For more information on Jupyter Notebooks visit the Jupyter project page at http://jupyter.org/.

Docker troubleshooting commands: (do not copy the commands, type them in the terminal)

To stop all containers:

```
docker kill $(docker ps -q)
```

remove all containers

```
docker rm $(docker ps -a -q)
```

remove all docker images

```
docker rmi $(docker images -q)
```

(The above commands can be modified to specific container ids and image ids.)

If you need to run the container but docker tells you the container name is already taken:

```
docker rm nameofcontainer
```

If you want to list containers and status:

```
docker ps -a
```

If you want to list all available docker images (no need to remove these - they are available but not necessarily running):

```
        docker images
```

If you want to work in docker from a standard terminal (vs. Jupyter):

```
docker exec -it nwm-training bash
```