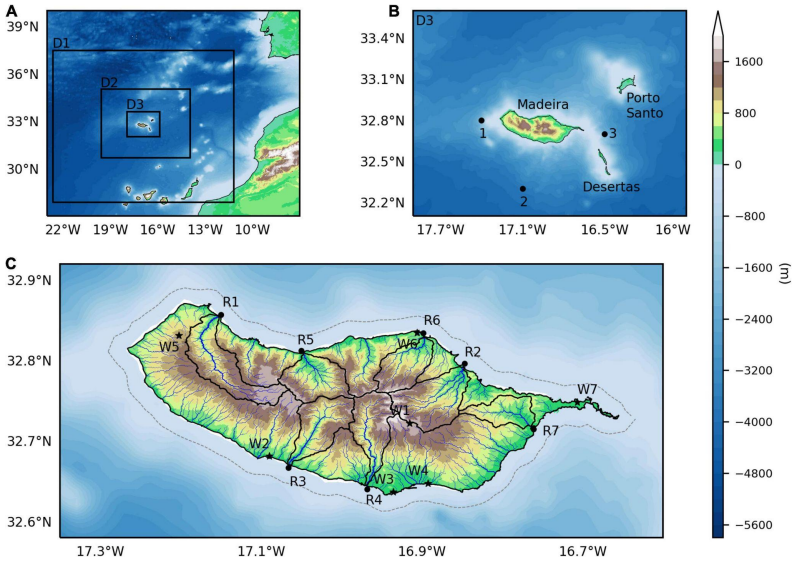
Weather Research & Forecasting Model (WRF)
Processing of forecasts with docker containers in Azure cloud environment

Observatório Oceânico da Madeira

Ricardo Faria
Luís Fernando Nóbrega



Environment setup

Observatório Oceânico da Madeira

Important aspects

- Used distribution: ubuntu 22.04;
- WRF Install Script: https://github.com/bakamotokatas/WRF-Install-Script/blob/master/README.md;
- Install script: WRF4.6.0_Install.bash;
- In order to save space while maintaining resolution, **GEOGRID.TBL** was altered to use 30s res as default and worse resolutions such as 2m, 5m, 10m , 1 deg and 2 deg were manually **deleted**;
- Image name: **reduced_ubuntu_image**;
- Image disk usage: 32.4 GB;
- tar.gz disk usage: 3.3 GB;
- Uncompressing time: Around **15 min** for decompression of *tar.gz

Docker container

Observatório Oceânico da Madeira

Docker file structure:

```
FROM ubuntu:22.04

LABEL Ricardo Faria <email@gmail.com>

ENV DEBIAN_FRONTEND=noninteractive \
        LANG=en_US.UTF-8 \
        LANGUAGE=en_US:en \
        LC_ALL=en_US.UTF-8

RUN apt-get update && \
        apt-get install -y \
        sudo \
        file \
        nano \
        locales \
        python3 \
        hostname \
        m4 \
        make \
        perl \
        tar \
        bash \
        tcsh \
        time \
        wget \
        cmake \
        pkg-config \
        libxml2-dev \
        libcurl4-openssl-dev \
        libnetcdf-dev && \
        apt-get clean && \
        rm -rf /var/lib/apt/lists/* && \
        locale-gen en_US.UTF-8 && \
        update-locale LANG=en_US.UTF-8 LC_ALL=en_US.UTF-8 && \
        useradd -m swe && echo "swe ALL=(ALL) NOPASSWD:ALL"
        > /etc/sudoers.d/swe && \
                echo "alias cp='cp -iv'" >> /home/swe/.bashrc && \
                echo "alias mv='mv -iv'" >> /home/swe/.bashrc && \
                echo "alias mkdir='mkdir -pv'" >> /home/swe/.bashrc &&
        \
                echo "alias ll='ls -FGlAhp'" >> /home/swe/.bashrc

USER swe
WORKDIR /home/swe
CMD ["bash"]
```

Only one **RUN** to reduce number of layers and occupied space

Docker container

Observatório Oceânico da Madeira

Other actions performed after docker initiation

- apt install pip
- pip install requests
- Manually set ln -s ungrib/Variable_Tables/Vtable.GFS Vtable for first time
- Same for ls -ls geogrid/GEOGRID.TBL

Resolution limitations (30s)

- **Deleted**: varsso_10m, varsso_5m, varsso_2m, orogwd_2deg, orogwd_1deg, orogwd_30m, orogwd_20m, lai_modis_10m;

- **Kept**: albedo_modis lai_modis_30s modis_landuse_20class_30s_with_lakes soiltemp_1deg soiltype_top_30s varsso greenfrac_fpar_modis maxsnowalb_modis orogwd_10m soiltype_bot_30s topo_gmted2010_30s

To add other resolutions: source1 or source2 in WPS_GEO directory

Docker container commands

Observatório Oceânico da Madeira

docker images	Lists all available images, size, ID, creation date and reposioty
docker ps -a	Lists all containers, exit status, creation date and container ID
docker run -it my_container_name	Starts a container in interactive mode. Only practical way to alter container files and internal architecture
docker commit container_id image_name	Allows altering of base image after exiting from container and retrieving container ID. Without it, changes remain local inside container
exit (from inside container)	To exit container from interactive mode. Should be followed by docker ps -a to check status of exit, which should be 0 for saves to be made
docker system df	Displays disk usage from whole docker environment
docker image prune	Removes dangling image creation layers -> **USE WITH CAUTION**
docker stop my_container	Stops containers running (from the outside)
docker remove my_container	Removes stopped containers

Editing the container interactively

Observatório Oceânico da Madeira

```
fernando@DESKTOP-L4H7EVR:~/OOM/copernicus3_images$ docker images        1 - choose image
REPOSITORY              TAG       IMAGE ID       CREATED        SIZE
reduced_ubuntu_image    latest    0c67e48bfa8b   6 days ago     32.4GB
my-ubuntu-22.04-image   latest    db73bb86acd9   10 days ago    65.9GB
fernando@DESKTOP-L4H7EVR:~/OOM/copernicus3_images$ docker run -it reduced_ubuntu_image   2 - run interactive mode
swe@126a61b4df09:~$ cd Build_WRF/
swe@126a61b4df09:~/Build_WRF$ ls
CONFIGS  LIBRARIES  WPS-4.6.0  WPS_GEOG  WRF-4.6.0-ARW      3 - naviage dirs and change what is needed
swe@126a61b4df09:~/Build_WRF$ cd CONFIGS/
swe@126a61b4df09:~/Build_WRF/CONFIGS$ ls
forecast_download.py  forecast.sh  historic_download.py  instructions.txt  model_set.py  namelist_editer.py  processors.py  wps_input.txt  wrf_input.txt
swe@126a61b4df09:~/Build_WRF/CONFIGS$ exit   4 - leave interactive mode
exit
fernando@DESKTOP-L4H7EVR:~/OOM/copernicus3_images$ docker ps -a   5 - see available containers to fetch ID
CONTAINER ID   IMAGE                  COMMAND   CREATED          STATUS                      PORTS     NAMES
126a61b4df09   reduced_ubuntu_image   "bash"    18 seconds ago   Exited (0) 3 seconds ago              inspiring_chaplygin
fernando@DESKTOP-L4H7EVR:~/OOM/copernicus3_images$ docker commit 126a61b4df09 my-ubuntu-22.04-image   6 - fetch changes to new image
```

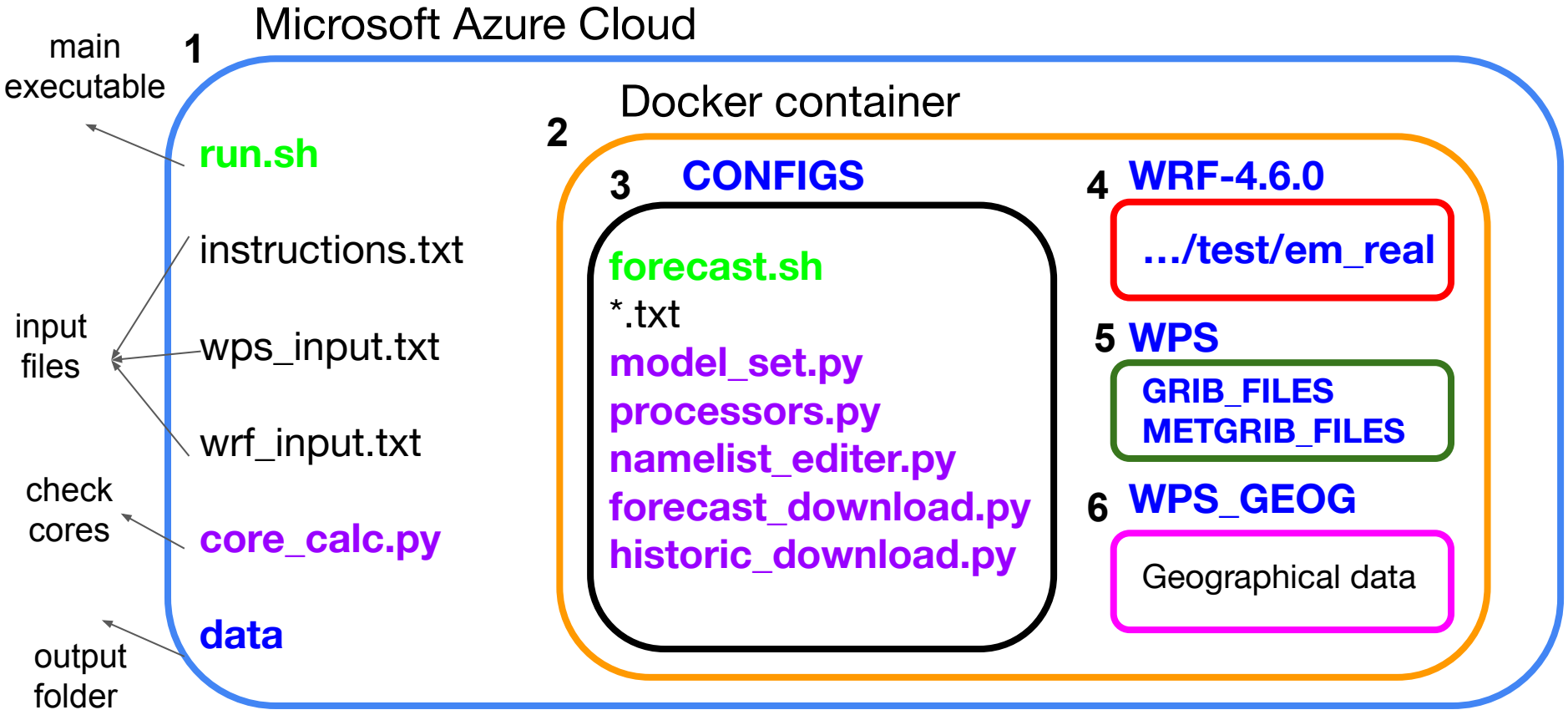
An id: usually something like sha:12@asd4$%39 will appear -> use $ docker images to check for updated date

If docker seems to take too much space:

DON'T use -a flag after $ docker image prune it deletes images without associated containers

```
fernando@DESKTOP-L4H7EVR:~/OOM/copernicus3_images$ docker system df
TYPE            TOTAL     ACTIVE    SIZE      RECLAIMABLE
Images          2         0         98.29GB   98.29GB (100%)
Containers      0         0         0B        0B
Local Volumes   0         0         0B        0B
Build Cache     23        0         446.7MB   446.7MB
fernando@DESKTOP-L4H7EVR:~/OOM/copernicus3_images$ docker image prune
```

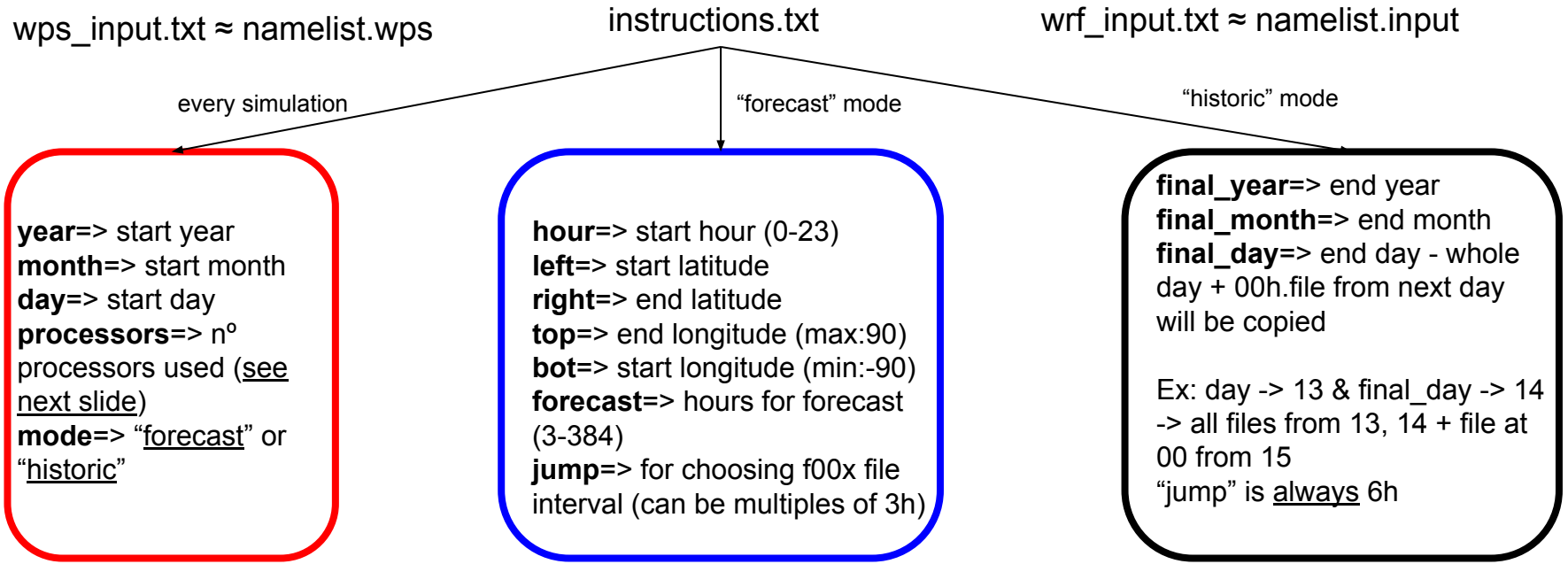
Running Environment structure



Microsoft Azure cloud

Observatório Oceânico da Madeira

Necessary input files before ./run.sh



Notes: The simulation working directory **MUST** have a instructions.txt similar to the provided one .The files should be manually revised before ./**run.sh** .All terms are **CASE SENSITIVE** and should not have any capitalization. **DO NOT** change the **Bold** terms.

Microsoft Azure cloud

Observatório Oceânico da Madeira

Example input.txt

```
############## READ ME
# forecast data from "Data Transfer: NCEP GFS Forecasts (0.25 degree grid)" -> Data until 10days back
# most recent forecast data available -> 6h before current date (updated at 6,12,18,24 h) -> max forecast -> 384h -> aprox 14days
#
# historic data from "NCEP GDAS/FNL 0.25 Degree Global Tropospheric Analyses and Forecast Grids" -> Data until 10 years back
# most recent historic data available -> 48h before current date
############## For both setups (historic or forecast)

year=2020
month=8
day=13

processors=4
mode=historic

############## For forecast
# left & right -> latitude (min:0 max:360)
# top & bot -> longitude (min:-90 max:90) -> top > bot

hour=0
left=250
right=360
top=40
bot=-5
forecast=3
jump=3

############## For historic
# final_day -> whole day + 00h.file from next day will be copied
# Ex: day -> 13 & final_day -> 14 -> all files from 13, 14 + file at 00 from 15
# for data of just 1 entire day -> day same as final_day
# hour_jump is always 6h in this mode

final_year=2020
final_month=8
final_day=13
```

For multiple domains, the simulation period <u>will be the same</u>. Coordinates should accommodate the <u>largest</u> domain

These must include **ref_lat** and **ref_lon** that is in wps_input.txt

This file changes automatically, however **make sure this exists** before running anything

Microsoft Azure cloud

Observatório Oceânico da Madeira

Choosing appropriate number of processors

core_calc.py -> interactive function to check appropriate number of cores to be used. Checks wps_input.txt and retrieves a core interval. It **DOES NOT** change the value in instructions.txt but stops program in case of excessive nº cores -> returns MIN processors: # and MAX processors: # as well as a **warning** if size and time parameters of **first domain** do not match in wrf_input.txt, wps_input.txt and instructions.txt

For your **smallest-sized** domain:
((e_we)/25) * ((e_sn)/25) = most amount of processors you should use

For your **largest-sized** domain:
((e_we)/100) * ((e_sn)/100) = least amount of processors you should use

If simulation is **failing**, usually use **more** processors!

For more information: Number_of_cores

core_calc.py always runs before **run.sh** to prevent mismatches or missing files

Microsoft Azure cloud

Observatório Oceânico da Madeira

Starting Azure VM instance

```
# Log in to Azure
az login

# Create a resource group if you don't have one
az group create --name myResourceGroup --location eastus

# Create the virtual machine
az vm create \\
  --resource-group myResourceGroup \\
  --name myVM \\
  --image UbuntuLTS \\
  --size Standard_D32ls_v5 \\
  --admin-username azureuser \\
  --generate-ssh-keys
```

```
# Save the Docker image locally
docker save -o mydockerimage.tar mydockerimage:latest

# Transfer the image to the Azure VM
scp mydockerimage.tar azureuser@<vm-public-ip>:/home/azureuser/
```

```
docker run -d --name mycontainer -e START_DATE=2023-01-01 -e END_DATE=2023-12-31 \\
  -v /mnt/data:/app/data myContainerRegistry.azurecr.io/mydockerimage:latest
```

```
# Stop the VM
az vm stop --resource-group myResourceGroup --name myVM

# Optionally, delete the VM and associated resources
az vm delete --resource-group myResourceGroup --name myVM --yes --no-wait
```

Check the Logs:

```
docker logs mycontainer
```

Download Data:

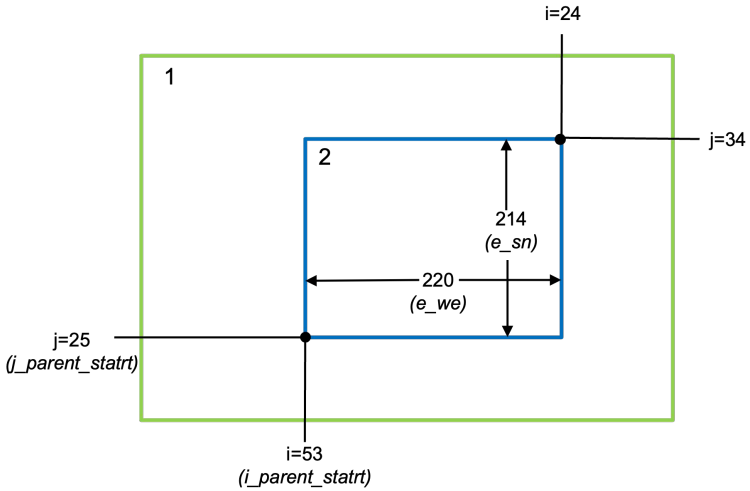
```
scp azureuser@<vm-public-ip>:/path/to/data /local/path/
```

Microsoft Azure cloud

Observatório Oceânico da Madeira

Notes about Nested Domains

- Geogrid is very sensitive to entry params;
- Nº domains is set with **max_dom** = 1;
- e_we = 100,46, and e_sn = 100,37,
- (e_sn−s_sn+1)= m× parent_grid_ratio+1 (m is an integer);
- In wrf.input **topo_wind** = 1, 0, 0, might need to be set to 0 in nested domains due to insufficient data;
- **i_parent_start** and **j_parent_start** define the domains position. If nested domain is outside parent domain, make sure to reduce size or change the i and j start tiles.
- Alternatively, increase **parent_grid_ratio**



Microsoft Azure cloud

Observatório Oceânico da Madeira

Running command

./run.sh -e START_DATE=2023-01-01 -e END_DATE=2023-01-02

- START_DATE and END_DATE **must** be provided even if files are correctly set;
- Both environment variables can be in **%Y-%m-%d** or **%Y-%m-%d_%H:%M:%S**;
- Even though the format asks for minutes and seconds, they have to be **00:00** as the data source only contains hourly files.
- By default, if **%Y-%m-%d** is provided, it will auto fill it with **00:00:00**;
- Before proceeding a input will be required -> **CHECK** everything before running.
- **Warnings** might be given -> Some are normal depending on the format of instructions;
- Ex: WARNING: wps and wrf input files do not have the same dx -> if multiple domains with different sizes are used, this will appear for sure;
- For historic mode, choose periods that are multiples of 24h.

Docker container

Observatório Oceânico da Madeira

forecast.sh -> has all instructions for executing the remaining files, geogrid, ungrib, metgrib, run.exe and wrf.exe. Almost error messages related with missing files or subprocesses are here. Check example file for unbugging.

*.txt -> all txt files that were copied from Azure cloud and will change inside files in WPS and WRF dirs

model_set.py -> reads instructions.txt to determine which mode to run, “forecast” or “historic” -> case sensitive

processors.py -> reads instructions.txt to determine number of processors to use -> case sensitive

namelist_editer.py -> changes namelist.input and namelist.wps with the .txt provided in Azure cloud

forecast_download.py -> checks connection and downloads grib files for “forecast” mode. Will search based on terms such as “top”, “bot”... etc -> IF download fails, either URLS changed and file must be changed manually or too big of a domain was chosen. If grib files exceed 1GB, URL request error may occur -> check log for errors https://nomads.ncep.noaa.gov/cgi-bin/filter_gfs_0p25_1hr.pl

historic_download.py -> checks connection and downloads fnl files for “historic” mode. Unlike forecast, the files will always include the whole world and not a snippet domain. https://rda.ucar.edu/datasets/d083003/dataaccess/#

Debugging and common errors

Observatório Oceânico da Madeira

- **forecast.sh instantly failed** -> incorrect format for input files

- **Download failed or taking too long** -> domain choice too big; grib files may be too big; URL provider might have changed (check demo links); time period might be too big; dates in instructions.txt might not be correct; Check if wrong operating mode was chosen “forecast” or “historic”; Internet connection might be too slow.

- **Geogrid failed** -> incorrect info in wps_input (maybe in instructions.txt)
- **Ungrib failed** -> incorrect info in wps_input and instructions.txt; dates not coincident; date in instructions does not have all necessary points referenced in wps_input -> missing files; files might be too small due to download failure -> check size with “ll” in bash; files might be incomplete for performed tasks; with multiple domains, remember that the start and end dates must be the same
- **Metgrib failed** -> incorrect info in wrf_input and wps_input -> check compatibility and recheck instructions.txt; historic files might not contain all needed info for metgrid if they are too old; Chosen domain and coordinates might not be the same as the wps_input ones -> check if ref_lat and ref_lon are within domain chosen for download data

- **Run.exe or wrf.exe failed** -> Insufficient data; Invalid time period etc… -> eventual failures usually occur before this stage

- **files not saved to /data** -> check console log for wrf.exe errors; chosen simulation time period might be too small

For further information about the process: Compilation; running_WPS; WPS_namelist_variables; running_WRF; general_info