

Manual - How to use WRF with EMS

on Physon cluster

Tzvetan Simeonov

July, 2015

Software	version
WRF	
EMS	

Installation

To be written

Starting a parallel session on Physon

First of all a parallel session should be initiated:

```
qsh -l h_rt=4:0:0,h_stack=512M -pe ipar 8 (1)
```

The `qsh` starts a interactive session on one of the cluster computation nodes, the `h_rt=4:0:0` sets the time limit of the session, the `h_stack=512M` sets the memory (RAM) limit (M for MB and G for GB) and the `-pe ipar 8` sets the number of requested cores. The WRF can run on 1 core, but multi-core processing is preferable.

The Physon clusters' architecture consists of 20 nodes with 8 cores each. A parallel session with 10 cores from two nodes may take longer time to process, than a session with 8 cores from one node due to architecture limitations. The optimal number of cores per session is a multiple product of 8.

Setting up domain with EMS

When already in the interactive parallel session, some folders and variables need to be set up for the model. Setting up the number of slots (needed for WRF):

```
export NSLOTS=8 (2)
```

Setting up the working directory:

```
export MYWORKDIR=.../wrf (3)
```

Creating working subdirectories:

```
mkdir -p $MYWORKDIR/runs (4)
```

```
mkdir -p $MYWORKDIR/logs (5)
```

Initializing the created directories:

```
export EMS_RUN=$MYWORKDIR/runs (6)
```

```
export EMS_LOGS=$MYWORKDIR/logs (7)
```

Before initiating computations on the Physon, some modules have to be loaded:

```
. /opt/Modules/modules.sh (8)
```

The EMS.profile module should be loaded:

```
. /opt/physon/ems/wrfems/etc/EMS.profile (9)
```

After the loading the following response may follow:

```
bash: ulimit: stack size: cannot modify limit: Operation not permitted
```

It is to be ignored.

The command to start the interactive domain setting is:

```
dwiz (10)
```

If the program response is as follows:

```
EMS_RUN = /work/x04/csimeonov/physon/runs
Starting EMS routine dwiz (V3.4.1.15.16) on cn003 at Tue Jul 7 12:36:13
2015 UTC
sh: /opt/physon/ems/wrfems/logs/dwiz.log: Permission denied
Domain Wizard Information has been written to /opt/physon/ems/wrfems/logs/dwiz.log
Alphanumeric code 6EQUJ5 will someday be interpreted as: "Think Globally,
Model Locally!"
```

Then execute 7 again.

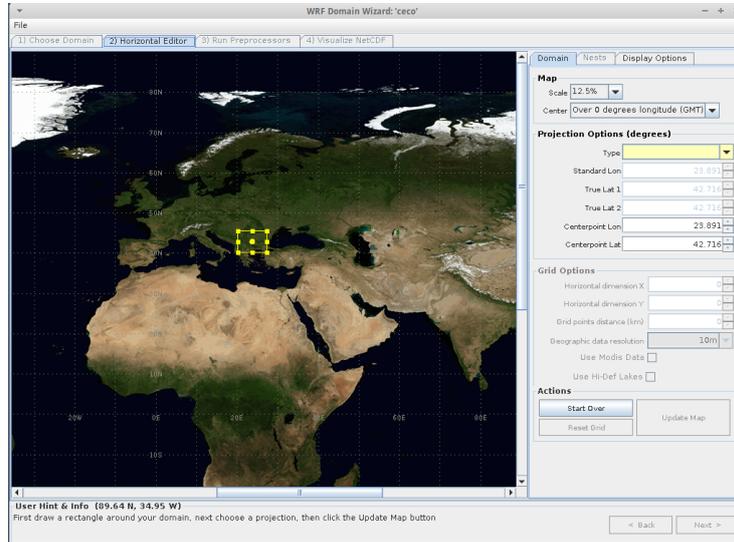
Once in the EMS GUI, the following window will appear:



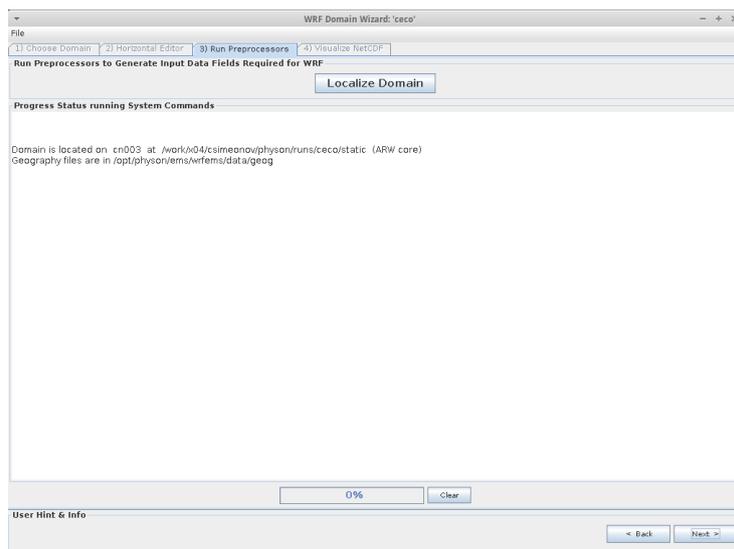
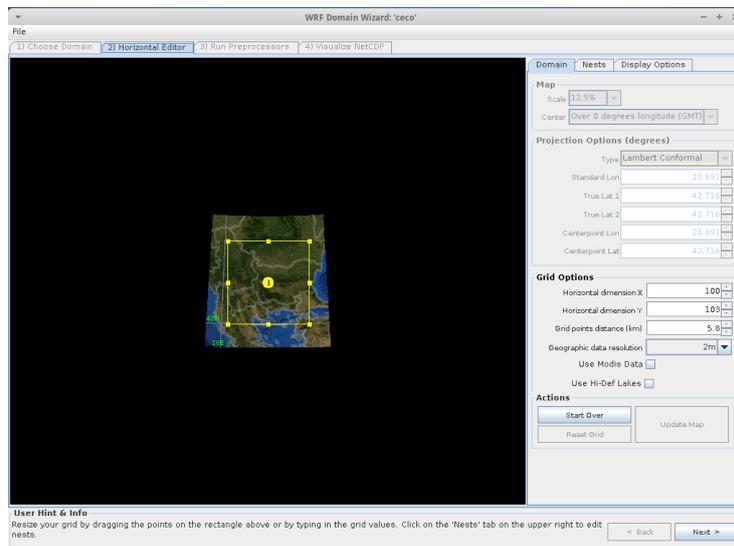
If you choose the **Create New Domain** option, the following windows will open:



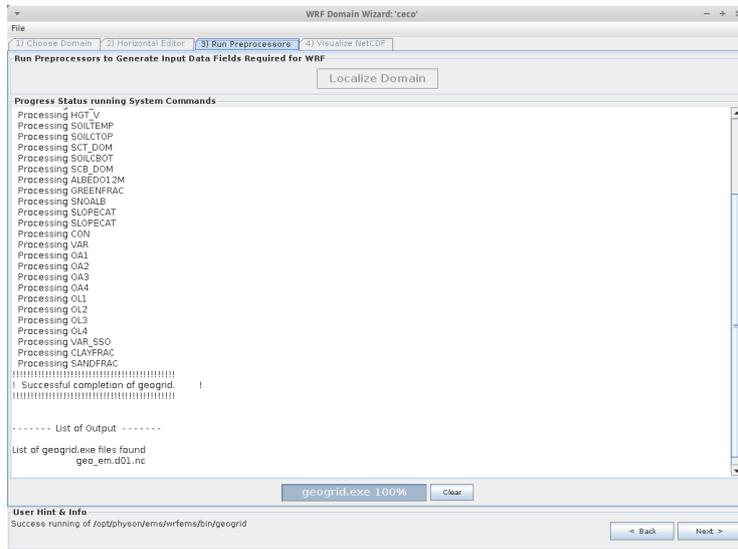
Insert the name of your domain and proceed to next step - choosing the center of the wanted domain:



After defining the model borders, the domain should be localized:



After the localization, the process is finished:



In case you want to use an already existing domain, you should choose the option in the beginning. Afterwards, the process is similar to the above mentioned steps



Starting a model run

First one should go to the domain directory. It is created in the previous step in case of a new domain:

```
cd $MYWORKDIR/runs/mydomain (11)
```

Afterwards the model is started using the following command:

```
./ems_autorun --domains 1 --length 3 (12)
```

Here in the `--domains 1,2` the used domains are specified. If a nested domain is included in the processing during the domain set-up step, than the `--domains 1,2` should be specified. If only a single domain was specified, `--domains 1` should be applied. The sting `--length 3` specifies the length of the run in hours.

The `./ems_autorun` is the simplest command to start the model run. If data assimilation is needed, a separate, more complicated procedure is to be made. The launched simulation will be for the current day and time. In order to change the date and time of the simulation ...

Post process the data

First in order to postprocess the generated data, a special directory has to be created, where the file `psadilookup.dat` has to be stored. The places, where the files are situated are specific for Physon=

```
cd $MYWORKDIR/postprocess (13)
```

```
mkdir -p $MYWORKDIR/postprocess/src (14)
```

```
cp /opt/physon/arwpost/3.1/data/psadilookup.dat src/psadilookup.dat (15)
```

Then the `namelist.ARWpost` file should be copied into the wanted directory for postprocessing.

```
cp /opt/physon/arwpost/3.1/data/namelist.ARWpost_template namelist.ARWpost (16)
```

The `namelist.ARWpost` should be edited with replacing the next list of values: `start_date=` and `end_date=` to be specified with the processed dates; `input_root_name=/runs/mydomain/wrfprd/wrfout_d01` is used to show where the output binaries from the WRF run are situated;

After performing this changes to the ARW settings file, the ARWpost program should be executed (but only after loading the modules on Physon):

```
module load arwpost (17)
```

```
ARWpost.exe > arw.log (18)
```

Displaying data

Grads is a program, designed for representing meteorological, geophysical and geodetic data on maps of the Earth. The program is started using the

```
grads (19)
```

command on Physon. The program uses a terminal interface. In order to visualize data, the generated data file from ARWpost should be openedL

```
open extract.ctl (20)
```

Then the high resolution maps can be loaded:

```
set mpdset hires (21)
```

In order to plot pressure from the file, the following commands should be executed:

```
set t 1 (22)
```

```
d pressure (23)
```

The other fields are defined as `wspd` for wind speed, `t2` for temperature at 2m above surface and many more. All the fields could be found in the `fields =` line of the `namelist.ARWpost` file.

For more information on how grads works, visit this page: <http://iges.org/grads/gadoc/users.html>