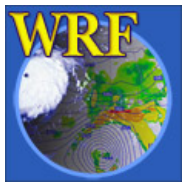
How to Set Up and Run WRF (*real.exe* & *wrf.exe*)?

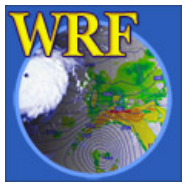
Wei Wang

July 26, 2005



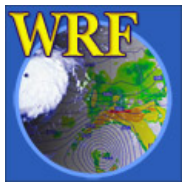
Outline

- How to download, compile and run WRF code?
- Input and output files
- Check output
- WRF namelist (runtime options)



How to download and compile WRF?

- Download WRF source code from
  http://www.mmm.ucar.edu/wrf/users/downloads.html
- What you will (in a week or so) get is
  `WRFV2.1.TAR.gz`
  For the practice, get the file from our local disk
- After **`gunzip`** and **`untar`**, you should see a directory **`WRFV2/`**
- cd to **`WRFV2/`** directory, and you should see ..



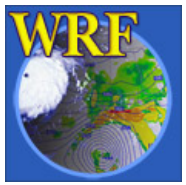
WRFV2 Directory

```
Makefile
README
README_test_cases
clean
compile
configure
Registry/
arch/
dyn_em/
dyn_exp/
external/
frame/
inc/
main/
phys/
share/
tools/
run/
test/
```

compile scripts

code directories

run directories



How to Compile

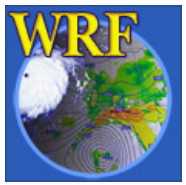
Type

./configure

Hint: If you need to define the path to netCDF, do it before typing **'configure'** using the environment variable NETCDF. e.g. on NCAR's IBM

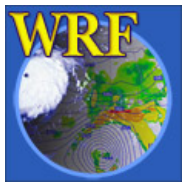
```
setenv NETCDF /usr/local/lib64/r4i4
```

Hint: If you use a Linux, make sure your netCDF library is installed using the same compiler you use to compile WRF



How to Compile

```
checking for perl5... no
checking for perl... found /usr/local/bin/perl (perl)
Will use NETCDF in dir: /usr/local/netcdf
PHDF5 not set in environment. Will configure WRF for use without.
------------------------------------------------------------------------
Please select from among the following supported platforms.
   1.  Compaq OSF1 alpha          (single-threaded, no nesting)
   2.  Compaq OSF1 alpha          (single-threaded, nesting using RSL without MPI )
   3.  Compaq OSF1 alpha SM       (OpenMP, no nesting)
   4.  Compaq OSF1 alpha SM-Parallel       (OpenMP, nesting using RSL without MPI )
   5.  Compaq OSF1 alpha DM-Parallel       (RSL_LITE, MPICH, allows nesting)
   6.  Compaq OSF1 alpha DM-Parallel       (RSL, MPICH, allows nesting)
   7.  Compaq OSF1 alpha DM-Parallel/SM-Parallel    (RSL, MPICH, OpenMP, allows nesting)
   8.  Compaq OSF1 alpha DM-Parallel/SM-Parallel    (RSL, DECMPI, OpenMP, allows nesting)
   9.  Compaq OSF1 alpha DM-Parallel       (RSL, MPICH, MCEL EXPERIMENTAL)
  10.  Compaq OSF1 alpha DM/SM    (BUILD FOR AUTODOC ONLY -- DO NOT USE TO COMPILE CODE)
Enter selection [1-10] :
```



How to Compile

This will create a file called

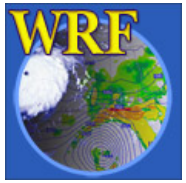
`configure.wrf`

Hint: You can edit this file to change compile option

Hint: If compile options are not provided for your system. You may edit

`arch/configure.defaults`

to add a new option.



How to Compile

Optionally if one would like to use the moving nest option, edit

`configure.wrf`

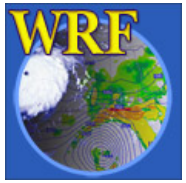
and add

`-DMOVE_NESTS` (for specified move) or

`-DMOVE_NESTS` and **`-DVORTEX_CENTER`**

(for automatic vortex-vortex following moving nest)

to ARCHFLAGS



How to Compile

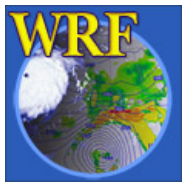
Type one the following command to compile:

```
compile
compile em_real
compile em_b_wave
compile em_hill2d_x
compile em_quarter_ss
compile em_squall2d_x
compile em_squall2d_y
compile em_grav2d_x
```

→ List of options

Create ideal.exe or real.exe, and wrf.exe

Typing one of the above will produce both initialization and model executables



How to Compile

Type **`compile em_real`**

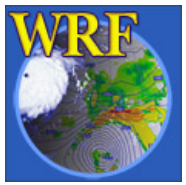
If compile is successful, you should have three executables built in directory main/:

`real.exe`

`ndown.exe` (for one-way nest processing)

`wrf.exe`

And these executables are linked to **`./test/em_real/`** and **`./run/`** directories



How to Compile

If you type

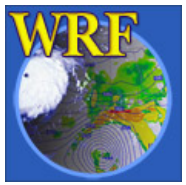
`compile em_quarter_ss`

you'll have

`ideal.exe`

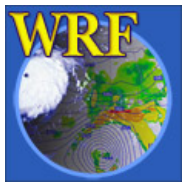
`wrf.exe`

linked to **`./test/em_quarter_ss/`** and **`./run/`** directories



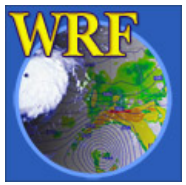
WRFV2/test directory

```
em_b_wave/
em_grav2d_x/
em_hill2d_x/
em_quarter_ss/
em_real/
em_squall2d_x/
em_squall2d_y/
```



WRFV2/run directory

```
LANDUSE.TBL
ETAMPNEW_DATA
RRTM_DATA
SOILPARM.TBL
VEGPARM.TBL
tr49t67
tr49t85
tr67t85
gribmap.txt
namelist.input          - require editing
real.exe -> ../main/real.exe
wrf.exe -> ../main/wrf.exe
ndown.exe -> ../main/ndown.exe
```



namelist.input

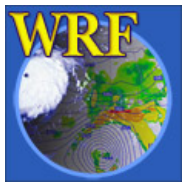
Before running **`real.exe`** or **`ideal.exe`**, and **`wrf.exe`**, edit **`namelist.input`** file for runtime options

A more extensive list of namelist and their explanations can be found in

`WRFV2/run/README.namelist,`

and in the User's Guide (p5-21 – 5-32)

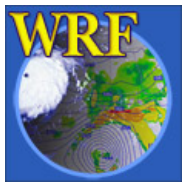
The complete list of available namelist options and default values are in **`WRFV2/Registry/Registry.EM`**



Sample of Registry.EM

Example 1: time control

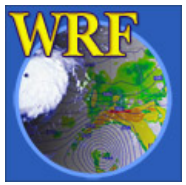
```
rconfig integer history_interval_mo namelist,time_control
   max_domains 0 h "history_interval_mo" "" "MONTHS"
rconfig integer history_interval_d namelist,time_control
   max_domains 0 h "history_interval_d" "" "DAYS"
rconfig integer history_interval_h namelist,time_control
   max_domains 0 h "history_interval_h" "" "HOURS"
rconfig integer history_interval_m namelist,time_control
   max_domains 0 h "history_interval_m" "" "MINUTES"
rconfig integer history_interval_s namelist,time_control
   max_domains 0 h "history_interval_s" "" "SECONDS"
```



Sample of Registry.EM

Example 2: time control

```
rconfig integer history_begin_y namelist,time_control max_domains 0
   h "history_begin_y" "" "YEARS from start of run"
rconfig integer history_begin_mo namelist,time_control max_domains
   0 h "history_begin_mo" "" "MONTHS from start of run"
rconfig integer history_begin_d namelist,time_control max_domains 0
   h "history_begin_d" "" "DAYS from start of run"
rconfig integer history_begin_h namelist,time_control max_domains 0
   h "history_begin_h" "" "HOURS from start of run"
```



How to Run real.exe?

To run on single or OpenMP systems, type

`ideal.exe`

for a idealized case, and

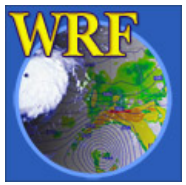
`real.exe`

for real-data cases.

To run on parallel system, typically type

`mpirun -np N real.exe`

where *N* is the number of processors



How to Run wrf.exe?

To run on single or OpenMP systems, type

`wrf.exe` or **`wrf.exe >& wrf.out`**

for all case, and on system using mpich, type

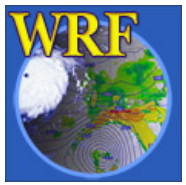
`mpirun -np N wrf.exe`

Hint: On IBMs, the MPI run command is

`poe wrf.exe` for batch job or

`poe wrf.exe -rmpool 1 -procs 4`

for interactive job



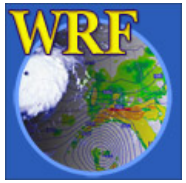
Output from mpirun

The standard-out and error will go into the following files for MPI runs:

`show_domain_0000:` domain-deposition info from RSL

`rsl.out.0000`	**`rsl.error.0000`**
`rsl.out.0001`	**`rsl.error.0001`**
`rsl.out.0002`	**`rsl.error.0002`**
`rsl.out.0003`	**`rsl.error.0003`**

There are one pair of files for each running processor



WRF Files

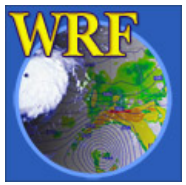
Input to **`real.exe`**: multiple files from SI

`wrf_real_input_em.d01.yyyy-mm-dd_hh:00:00`

Output from **`real.exe`**:

`wrfinput_d01` ← Single time level data

`wrfbdy_d01` ← BC data for multiple times



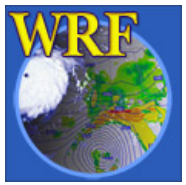
WRF Files

Output from wrf.exe (by default):

`wrfout_d01_ yyyy-mm-dd_hh:00:00`

If restart is activated:

`wrfrst_d01_yyyy-mm-dd_hh:00:00`



Check Output

Check run log file:

`wrf: SUCCESS COMPLETE WRF`

Use **`ncdump`**

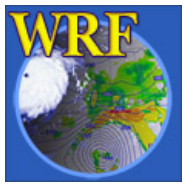
`ncdump -v Times wrfout_d01_*`

to check output times. Or

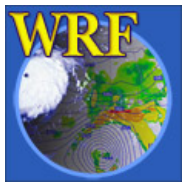
`ncdump -v U wrfout_d01_*`

to check a particular variable (U)

Use **`read_wrf_nc.f`** (see “Graphics Tools” lecture)



ARW
runtime options



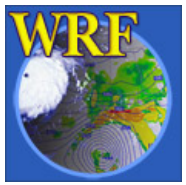
namelist.input

Six namelists:

- **`&time_control`**
- **`&domains`**
- **`&physics`**
- **`&dynamics`**
- **`&bc_control`**
- **`&namelist_quilt`**

As a general rule:

- Multiple columns: domain dependent
- Single column: value valid for all domains



&time_control

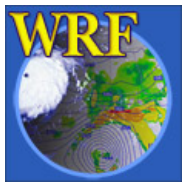
Run time control:

`run_days,run_hours, run_minutes, run_seconds` (WRF coarse grid only)

`start_year, start_day, start_hour, start_minute, start_second, end_year, end_day, end_hour, end_minute, end_second` (real and WRF, esp. for nest)

Input data interval control:

`interval_seconds` (real only)



&time_control

Output control:

`history_interval:` output frequency in minutes

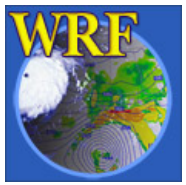
`frame_per_outfile:` used to split output files

`restart:` whether this is a restart run

`restart_interval:` used to write restart file

`io_form_history/restart/initial/boundary:`

IO format (mostly set to 2 for netCDF; Other options: 1 – binary; 4 – PHDF5; 5 – GriB 1)



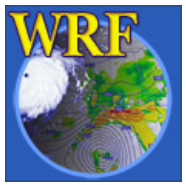
&time_control

Nest input control:

`input_from_file:` whether one would use wrfinput_d0n (n>1) as input.

`fine_input_stream:` how nest domain input are used: = 0 – all input used; = 2 – only static input and masked fields are used

Hint: **`fine_input_stream = 2`** option allows a nest to start at a later time



&domains

Time step control:

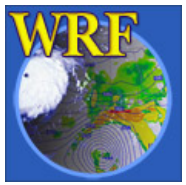
`time_step:` integer

`time_step_fract_num :` numerator for fractional time step

`time_step_fract_den:` denominator for fractional time step

Example: if one would specify time step of 15.5 sec, set

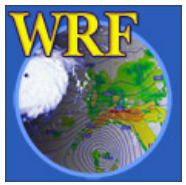
```
time_step = 15
time_step_fract_num = 1
time_step_fract_den = 2
```



&domains

Domain dimension control:

- **`s_we:`** always set to 1
- **`e_we:`** domain dimension in x direction (non-staggered)
- **`s_sn:`** always set to 1
- **`e_sn:`** domain dimension in y direction (non-staggered)
- **`s_vert:`** always set to 1
- **`e_vert:`** domain dimension in z (full η levels)
- **`dx, dy: (dx=dy)`** grid distance in meters
- **`ztop:`** only used in idealized case to set model top



&domains

Nest control:

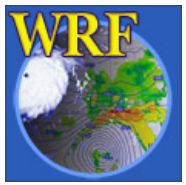
`max_dom:` how many domains to run

```
grid_id, parent_id, i_parent_start,
  j_parent_start, parent_grid_ratio,
  parent_time_step_ratio
```

```
feedback = 0, 1
```

```
smooth_option = 0, 1 or 2
```

(see “Real Data Initialization” lecture)



&domains

Moving nest control:

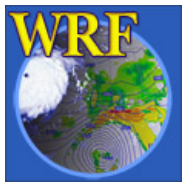
Two options available:

- specified move:

  num_moves, move_id, move_interval, move_cd_x, move_cd_y

- automatic move: use a vortex-following algorithm

  vortex_interval (default 15 min)
  max_vortex_speed (default 40 m/s)
  corral_dist (default 8 coarse grid cells)



&physics

Seven major physics categories:

`mp_physics: 0,1,2,3,4,5,6,98,99`

`ra_lw_physics: 0,1,99`

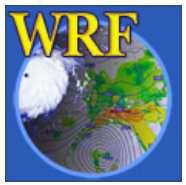
`ra_sw_physics: 0,1,2,99`

`sf_sfclay_physics: 0,1,2` (set before running read or ideal)

`sf_surface_physics: 0,1,2,3`

`bl_pbl_physics: 0,1,2,99`

`cu_pbl_physics: 0,1,2,3,99`



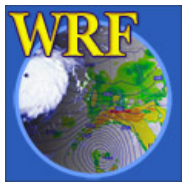
&physics

Physics call time control:

- **`radt:`** for radiation calls
- **`bldt:`** for surface and PBL calls
- **`cudt:`** for cumulus calls

Negative moisture variable control:

- **`mp_zero_out`**: 0, 1, or 2
- **`mp_zero_out_thresh`**: 1E-8



&physics

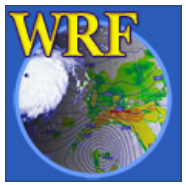
Other useful ones:

`surface_input_source:` whether to use WRFSI landuse and soil cat data, or from GriB file

`num_soil_layers:` different values for different sf_surface_physics options

Sea-ice temperature control:

`seaice_threashold:` 271 K (default)



&physics

Sea-surface temperature update control:

`sst_update:` 0 – no SST update

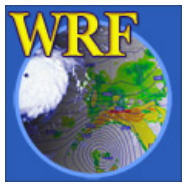
1 – update SST

Set before running **`real.exe`**, and this will create an additional output from **`real.exe`**: wrflowinp_d01

To use the file in **`wrf.exe`**, in **`&time_control`**, add

`auxinput_inname` = “wrflowinp_d01”

`auxinput_interval` = 360



&physics

Sensitivity tests:

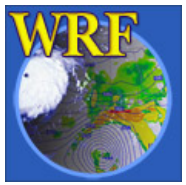
`isfflx:` 0, or 1

`icloud:` 0, or 1

Grell-Devenyi cumulus scheme control:

`maxiens, maxens, maxens2, maxens3:`

ensemble member dimensions for multiple closures and multiple parameter controls. Leave them as they are.



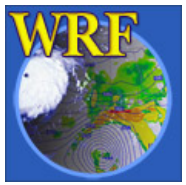
&dynamics

Diffusion/filter options:

`diff_opt, km_opt:` typically not required when dx > 10 km

`w_damping:` real-time only, used to control excessive vertical motion

`damp_top, zdamp, dampcoef:` mostly used in idealized simulations. Do not work for real-data cases. In real-data cases, ptop is recommended to be placed at least at 50 mb (or ~ 20 km)



Recommended Options for dx < 10 km

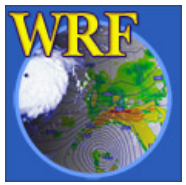
Some explicit diffusion is required, esp. under extreme convective conditions:

&dynamics

`diff_opt = 1`

`km_opt = 4`

`w_damping = 1` (for real-time runs)



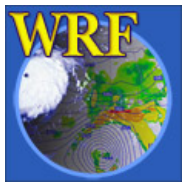
&dynamics

Other options:

`smdiv:` divergence damping control (~ 0.1)

`emdiv:` external mode control (~ 0.01)

`epssm:` coeff for vertically implicit off-centered acoustic step (~ 0.1)



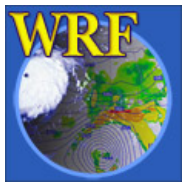
&dynamics

Base state parameter control:

`base_temp:` default value is 290 K

`base_pres:` default value is 100000 Pa

`base_lapse:` default value is 50 K from 1000 to 400 mb

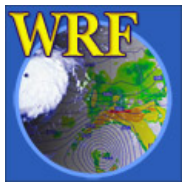


&dynamics

Other options:

`non_hydrostatic:` set to false to enable hydrostatic option

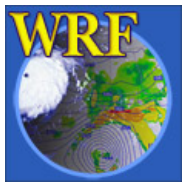
`time_step_sound:` may be altered when time step is very much larger than 6*DX



&bc_control

Four choices:

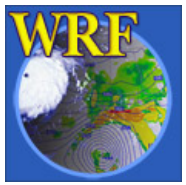
- **`Open_xs, open_xe`**
- **`symmetric_xs, symmetric_xe`**
- **`periodic_xs, periodic_xe`**
- **`Specified`** (real-data only, and set before running **`real.exe`**)
  - **`Spec_bdy_width:`** = spec_zone + relax_zone
  - **`spec_zone:`** = 1 (should not change)
  - **`relax_zone:`** = 4 (can be varied)



&namelist_quilt

Parallel I/O control:

`nio_tasks_per_group (>0):` allow IO to be done on separate processors



Recommended

Start with the namelist or namelists in a particular test directory, and make modifications.