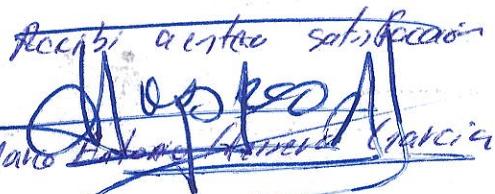


Manual Operativo

(Modelo WRF-CHEM para el estado de Puebla)



Recibi a estos sat. feccón
Mano de 

Contenido

Contenido

| | |
|---|----|
| 1. Introducción..... | 2 |
| 1.1 Objetivo | 2 |
| 1.2 Servidor y usuario..... | 2 |
| 1.3 Condiciones de Frontera | 6 |
| 1.4 Información geográfica estática..... | 8 |
| 2. Sistema de pre-procesamiento WRF (WPS)..... | 9 |
| 2.1 WPS..... | 9 |
| 2.2 Configuración del archivo de control namelist.wps | 10 |
| 2.3 Generando condiciones de frontera y geográficas con WPS | 15 |
| 3. Modelo WRF-Chem | 18 |
| 3.1 Sobre el modelo WRF | 18 |
| 3.2 Configuración de namelist.input para real.exe..... | 18 |
| Condiciones iniciales y de frontera | 21 |
| 3.3 Integración de emisiones al modelo WRF | 22 |
| 3.4 Configuración de namelist.input para wrf.exe | 24 |
| 3.4.1 Pronóstico con WRF-Chem | 26 |
| 4. Sistema autónomo para pronóstico meteorológico | 28 |
| 4.1 Elaboración autónoma de pronóstico | 28 |
| 4.1.1 Iniciar el sistema desde la terminal..... | 28 |
| 4.1.2 Tarea en crontab | 29 |
| 4.2 Problemas conocidos | 30 |
| Apéndice A | 33 |

1. Introducción

1.1 Objetivo

El presente documento tiene como finalidad listar las actividades y/o procedimientos para poner en operatividad el modelo WRF-Chem instalado en el servidor localizado en las instalaciones de **Secretaría del Medio Ambiente, Desarrollo Sustentable y Ordenamiento Territorial.**

1.2 Servidor y usuario

Para exemplificar algunos de los pasos para poner en operatividad el modelo WRF Chem se hará referencia a directorios existentes en la cuenta del servidor del usuario. El modelo se encuentra en la siguiente cuenta:

Dirección IP asignada: 10.0.20.60

El usuario en el cual se basa este documento es el siguiente:

Usuario: wrf1

Contraseña: wrf12345

Si se desea realizar pruebas con el modelo en cuanto al uso de las cuentas se recomienda crear una cuenta de **usuario** diferente donde el modelo y todas las paqueterías requeridas sean instalados. De preferencia este usuario debe estar agregado al grupo **sudo**. Otra opción es realizar una copia del directorio de trabajo del modelo WRF y trabajar en ella para evitar cambios en la versión instalada originalmente.

NOTA: En este documento los ejemplos de instrucciones en terminal de linux o rutas de directorios están marcados por el símbolo mayor que (>) y en color azul para separarlos del texto descriptivo. Si copia las instrucciones de este manual omita copiar dicho símbolo y sea atento en que la instrucción se haya copiado completa y sin espacios vacíos

Puede conectarse mediante la instrucción ssh de la siguiente manera en una terminal de linux o de su preferencia, como MobaXterm:

> **ssh -l wrf1 10.0.20.60**

Al conectarse se encontrará en el directorio home de su cuenta, puede asegurarse realizando los siguientes pasos.

> **cd**

> **pwd**

Si teclea los comandos anteriores verá que se debe situar en el directorio

> **/home/wrf1**

Una vez que se encuentre en la cuenta wrf1 podrá ver diferentes directorios, en seguida se listan los principales y una breve descripción de su contenido o función para familiarizarse con el entorno.

1. Directorio principal

> /home/wrf1/Workdir

Directorio de trabajo principal. En este directorio se encuentran todas las carpetas y subcarpetas de trabajo, como el modelo WRF, datos de condiciones de frontera, scripts de trabajo, etc.

2. Respaldos

> /home/wrf1/respaldos

Directorio de respaldo. En este directorio se encuentra un respaldo de los archivos de control para el funcionamiento del modelo (namelist.wps, namelist.input) entre otros. También en este directorio se encuentra un respaldo de los scripts principales para la operatividad del modelo, gráficas y respaldo de archivos comprimidos para reinstalar el modelo.

3. Directorio de ejecución del modelo

> /home/wrf1/Workdir/model-run

Directorio de operatividad del modelo. En este directorio se encuentran los programas elaborados para la operatividad autónoma del modelo, scripts de descarga de datos de frontera y archivos temporales para la operatividad del modelo, entre otros.

4. Directorio de programas principales

> /home/wrf1/Workdir/scripts

Directorio donde se encuentran los programas que realizan gráficos y condiciones de frontera de química atmosférica a partir de las observaciones de emisiones. En este directorio también se encuentra la carpeta *runs*, que es donde se guardan todas las salidas gráficas.

5. Salidas del modelo

> **/home/wrf1/Workdir/wrf_out**

En este directorio se envían las salidas del modelo y se guardan en formato NetCDF en un subdirectorio con la fecha y hora utc correspondiente de la corrida del modelo GFS (modelo que forza al modelo WRF).

6. WRF-Chem

> **/home/wrf1/Workdir/WRF-Chem**

Este es el directorio principal donde se encuentra el modelo WRF-Chem con su directorio WRF y WPS

7. Salidas GFS

> **/home/wrf1/Workdir/WRF-Chem/gfs**

Aquí se guardan los datos descargados del modelo global GFS que sirve como condición de frontera para el modelo WRF.

8. Emisiones

> **/home/wrf1/Workdir/emisiones**

Aquí se deben guardar las emisiones diarias observadas y que servirán como condiciones iniciales del módulo químico.

1.3 Condiciones de Frontera

Antes de iniciar los pasos de configuración y operatividad del modelo se deben descargar los datos que servirán como condiciones de frontera del modelo (forzamientos en las fronteras laterales del dominio o área de pronóstico). Para ello se ha desarrollado el programa en código bash script para la descarga de datos llamado *gfsrun.sh*, para esto debe posicionarse en el directorio model-run. Desde el directorio home se puede hacer de la siguiente manera.

> cd Workdir/model-run

En este directorio podrá encontrar el programa *gfsrun.sh* (puede usar *ls* en la terminal para poder verlo). Este programa fue diseñado para correrlo directamente en la terminal de linux y descargar datos del modelo GFS del sitio web <https://nomads.ncep.noaa.gov/>, de las 00:00Z horas o 12:00Z horas, dependiendo de la hora del día en el que se ejecute. Dentro de este archivo puede modificar el tamaño del dominio de trabajo (dominio de descarga) y directorios de trabajo, entre un par de opciones más. No se recomienda modificar parte del código en el programa original hasta tener una idea clara del funcionamiento. Para correr este programa en la terminal linux puede usar la siguiente instrucción:

> bash gfsrun.sh

Inmediatamente podrá ver en su pantalla la descarga de datos. Los archivos serán guardados en el directorio de Salidas del GFS. Cuando termine el programa le indicará y podrá verificar su descarga de la siguiente manera.

> ls -lh /home/wrf1/Workdir/WRF-Chem/gfs/yyyymmdd/hh

Donde yyyyymmdd y hh indican el año, mes, día y hora respectivamente correspondiente a la fecha en que se ejecutó el programa. Debe observar algo similar a la Figura 1.

```
(base) wrf1@srv1:~$ ls -lh WorkDir/WRF-Chem/gfs/20240404/00
total 405M
-rw-rw-r-- 1 wrf1 wrf1 16M Apr  4 04:25 gfs.20240404.00.000.grb
-rw-rw-r-- 1 wrf1 wrf1 16M Apr  4 04:25 gfs.20240404.00.003.grb
-rw-rw-r-- 1 wrf1 wrf1 16M Apr  4 04:26 gfs.20240404.00.006.grb
-rw-rw-r-- 1 wrf1 wrf1 16M Apr  4 04:26 gfs.20240404.00.009.grb
-rw-rw-r-- 1 wrf1 wrf1 16M Apr  4 04:26 gfs.20240404.00.012.grb
-rw-rw-r-- 1 wrf1 wrf1 16M Apr  4 04:27 gfs.20240404.00.015.grb
-rw-rw-r-- 1 wrf1 wrf1 16M Apr  4 04:27 gfs.20240404.00.018.grb
-rw-rw-r-- 1 wrf1 wrf1 17M Apr  4 04:27 gfs.20240404.00.021.grb
-rw-rw-r-- 1 wrf1 wrf1 17M Apr  4 04:28 gfs.20240404.00.024.grb
-rw-rw-r-- 1 wrf1 wrf1 17M Apr  4 04:28 gfs.20240404.00.027.grb
-rw-rw-r-- 1 wrf1 wrf1 17M Apr  4 04:29 gfs.20240404.00.030.grb
-rw-rw-r-- 1 wrf1 wrf1 16M Apr  4 04:29 gfs.20240404.00.033.grb
-rw-rw-r-- 1 wrf1 wrf1 16M Apr  4 04:29 gfs.20240404.00.036.grb
-rw-rw-r-- 1 wrf1 wrf1 17M Apr  4 04:30 gfs.20240404.00.039.grb
-rw-rw-r-- 1 wrf1 wrf1 17M Apr  4 04:30 gfs.20240404.00.042.grb
-rw-rw-r-- 1 wrf1 wrf1 17M Apr  4 04:30 gfs.20240404.00.045.grb
-rw-rw-r-- 1 wrf1 wrf1 17M Apr  4 04:31 gfs.20240404.00.048.grb
-rw-rw-r-- 1 wrf1 wrf1 17M Apr  4 04:31 gfs.20240404.00.051.grb
-rw-rw-r-- 1 wrf1 wrf1 17M Apr  4 04:31 gfs.20240404.00.054.grb
-rw-rw-r-- 1 wrf1 wrf1 17M Apr  4 04:32 gfs.20240404.00.057.grb
-rw-rw-r-- 1 wrf1 wrf1 17M Apr  4 04:32 gfs.20240404.00.060.grb
-rw-rw-r-- 1 wrf1 wrf1 17M Apr  4 04:33 gfs.20240404.00.063.grb
-rw-rw-r-- 1 wrf1 wrf1 17M Apr  4 04:33 gfs.20240404.00.066.grb
-rw-rw-r-- 1 wrf1 wrf1 17M Apr  4 04:33 gfs.20240404.00.069.grb
-rw-rw-r-- 1 wrf1 wrf1 17M Apr  4 04:34 gfs.20240404.00.072.grb
(base) wrf1@srv1:~$
```

Figura 1. Ejemplo de datos descargados con el programa gfsrun.sh, los archivos deben estar en formato grib.

Para ejecutar los programas en formato sh se recomienda usar en la terminal la instrucción bash, sin embargo no es limitativo al uso de sh, o punto barra.

1.4 Información geográfica estática

También será necesario que descargue solo una vez los datos de información geográfica con los que cuenta el modelo WRF en el siguiente sitio web:

WPS V4 Geographical Static Data Downloads Page

Esta información la usará el módulo WPS para las condiciones de suelo, relieve, entre otras características que el modelo WRF requiere para su funcionamiento. En el servidor se encuentra una copia completa en el directorio siguiente.

> **/home/wrf1/WorkDir/WRF-Chem/geog**

Sin embargo en caso de requerirse se sugiere descargar la base completa que se encuentra en línea para evitar problemas de funcionamiento del modelo.

2. Sistema de pre-procesamiento WRF (WPS)

2.1 WPS

En el directorio *WRF-Chem* se encuentra el módulo WPS, o sistema de procesamiento para el modelo WRF (WRF Preprocessing System). Aquí, el usuario puede definir el tamaño y número de dominios, preparar datos terrestres como topografía y tipo de suelo e interpolar los datos meteorológicos que provienen de las condiciones de frontera, en este caso del modelo GFS. Toda esta información se usará para ingresarla al modelo WRF.

> cd /home/wrf1/Workdir/WRF-Chem/WPS

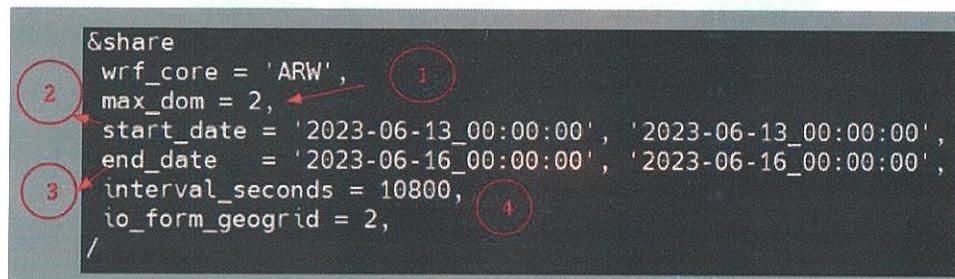
Si teclea los comandos anteriores verá que se debe situar en el directorio. En esta carpeta encontrará los siguientes archivos.

| Nombre del Programa | Descripción |
|---------------------|---|
| namelist.wps | Archivo de control |
| geogrid.exe | Programa que interpola las condiciones de topografía. |
| ungrib.exe | Programa que decodifica las condiciones de frontera del modelo GFS. |
| metgrid.exe | Programa que define los dominios e interpola al tamaño de la malla indicada en el archivo de control namelist.wps |

2.2 Configuración del archivo de control namelist.wps

El dominio requerido por el usuario, las fechas de pronóstico y resolución del modelo se configuran en el archivo namelist.wps, el archivo consiste en texto plano con una serie de opciones que se pueden editar. A continuación se detalla una a una de las opciones principales y elementales que puede cambiar. Si bien aquí se muestra a detalle los pasos que debe seguir para correr el modelo WRF se recomienda visitar el sitio web oficial¹ para ver el procedimiento de como ejecutar el modelo WRF-AWR que es la base de este manual.

La primera sección del archivo namelist.wps debe lucir como el mostrado en la Figura 2, y en la cual están señalados del uno al cuatro las opciones que debe modificar. En el Apéndice A se encuentra un ejemplo completo del namelist.wps así como en el directorio de respaldos.



```
&share
wrf_core = 'ARW',
max_dom = 2,
start_date = '2023-06-13_00:00:00', '2023-06-13_00:00:00',
end_date   = '2023-06-16_00:00:00', '2023-06-16_00:00:00',
interval_seconds = 10800,
io_form_geogrid = 2,
```

Figura 2. Ejemplo de sección de archivo namelist.wps. En este caso es similar al encontrado en el servidor. Hay una copia de respaldo en el directorio *respaldos*. Los campos señalados del 1 al cuatro son los correspondientes a editar.

EJEMPLO

Suponga el caso **arbitrario** que se han descargado datos con el programa gfsrun.sh para la fecha del 13 de junio del 2023 y desea obtener el pronóstico de las 12:00 UTC para las siguientes 72 horas en el estado de Puebla.

Paso 01: Definir el número de dominios que desea correr. Para este ejemplo por la cantidad de recursos de cómputo y área requerida se sugieren 2 dominios, el primero será el dominio con resolución más gruesa y el segundo corresponderá al dominio que cubre todo el estado de Puebla. Figura 2 opción 1.

Paso 02: Definir las fechas de inicio, las cuales deben corresponder con los datos iniciales del modelo GFS y se debe definir para el número de dominios que haya seleccionado. Figura 2 opción 2.

Paso 03: Definir las fechas de finalización del pronóstico. Debe estar seguro de contar con datos de condiciones de fronteras suficientes para el alcance de su pronóstico. Figura 2 opción 3.

Paso 04: En este campo debe indicar en segundos el tiempo en que al modelo WRF se le darán condiciones de frontera del modelo GFS. Figura 2 opción 4. En este ejemplo serán cada 3 horas.

A continuación se mostrará el proceso de configuración del tamaño del dominio en número de nodos de malla, Figura 3. Estas opciones son fijas, y no se deben cambiar cada ocasión que corra el modelo, excepto si se desea cambiar el dominio.

Paso 05: Punto donde inicia la malla, el primero siempre corresponde a 1, tanto el *j* como el *i* (*x* y *y*), mientras que la segunda columna corresponden al inicio del dominio dos.

```
&geogrid
  parent_id      = 1,   1,
  parent_grid_ratio = 1,   3,
  i_parent_start  = 1,   21,
  j_parent_start  = 1,   28,
  e_we            = 82,  82,
  e_sn            = 82, 103,
```

Figura 3. Ejemplo de sección de archivo namelist.wps. En este caso es similar al encontrado en el servidor. Hay una copia de respaldo en el directorio *respaldos*. Los campos señalados del 5 al cuatro son los correspondientes a editar.

Paso 06: Aquí corresponde al número de puntos de malla requeridos para cada dominio. Los mostrados en la Figura 3 opción 6 corresponden a los dominios operativos montados en el servidor.

Paso 07: Para el ejemplo del estado de Puebla con dos dominios se recomienda que el dominio mayor tenga una resolución de 12000 metros, así el dominio menor, el número dos que es el de interés, será de 4km (en una razón de 1:3 indicada en la opción parent_grid_ratio). Ésta opción se llena en los campos dx y dy como en la Figura 4. El tipo de proyección requerida para el estado de Puebla es mercator, debido a la latitud en la que se encuentra ubicada, otro tipo de proyección geográfica con los que cuenta el modelo WRF puede llevar al mal funcionamiento de éste.

```
geog_data_res = '30s', '30s',
dx = 12000,
dy = 12000,
map_proj = 'mercator',
ref_lat = 19.00,
ref_lon = -97.00,
geog_data_path = '/home/wrf1/WorkDir/WRF-Chem/geog/WPS_GEOG/'
/
```

Figura 4. Ejemplo de sección de archivo namelist.wps. En este caso es similar al encontrado en el servidor. Hay una copia de respaldo en el directorio *respaldos*.

En esta sección también se debe indicar la referencia geográfica (punto central) de la región deseada, en este ejemplo se sugiere que se indique la Latitud 19N y Longitud 97W.

Para verificar su dominio geográfico se recomienda usar una de las utilidades en el mismo módulo WPS, para ello necesitará tener instalado el lenguaje NCL.

En el servidor actual se encuentra la opción de activar este lenguaje mediante la instrucción.

> **conda activate ncl_stable**

Use la siguiente instrucción, y posteriormente debe arrojarle una imagen como la Figura 5 si es que los campos han sido llenados adecuadamente en el archivo namelist.wps.

> ncl util/plotgrids_new.ncl

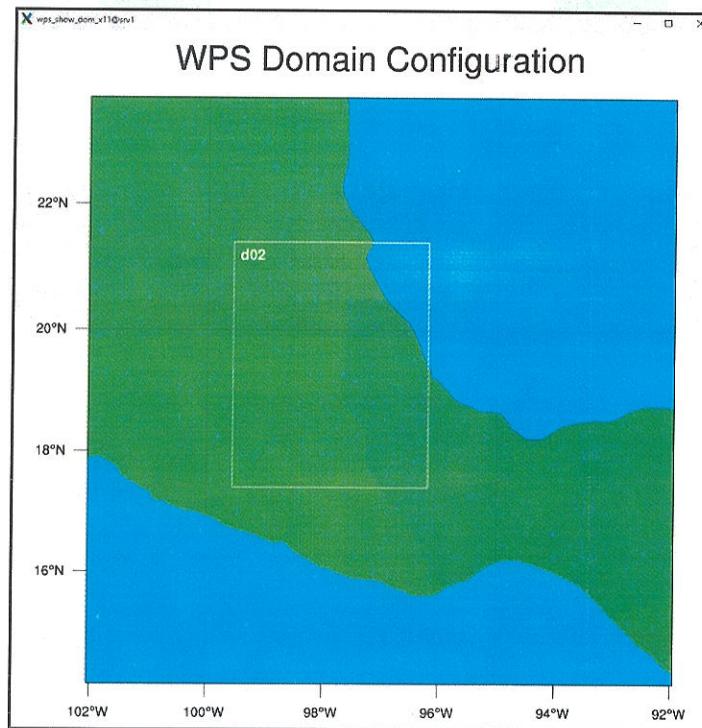


Figura 5. Ejemplo de figura de dos dominios.

2.3 Generando condiciones de frontera y geográficas con WPS

Una vez que se ha llenado adecuadamente el archivo namelist.wps debe continuar con la ejecución del programa geogrid.exe, ungrid.exe y metgrid.exe que se encuentran en el directorio.

Primero debe correr el programa geogrid.exe de la siguiente manera:

> ./geogrid.exe

Si todo se ejecutó de manera correcta debe obtener en su terminal un mensaje similar al siguiente:



```
!!!!!!  
! Successful completion of geogrid. !  
!!!!!!
```

Imagen ilustrativa

Posteriormente debe ligar los archivos grib que se descargaron como condiciones de frontera del modelo GFS.

> ./link_grib.csh ~wrf1/WorkDir/WRF-Chem/gfs/yyyymmdd/hh/

Esto le debe generar una serie de archivos con la estructura similar a GRIBFILE.AAA. Ahora puede ejecutar el programa ungrid.exe.

> ./ungrib.exe

Si todo se ejecutó de manera correcta debe obtener en su terminal un mensaje similar al siguiente:



Imagen ilustrativa

Esto le debe generar una serie de archivos con la estructura similar a FILE:2023-06-13_12. Ahora puede ejecutar el programa metgrid.exe.

> ./metgrid.exe

Si todo se ejecutó de manera correcta debe obtener en su terminal un mensaje similar al siguiente:



Imagen ilustrativa

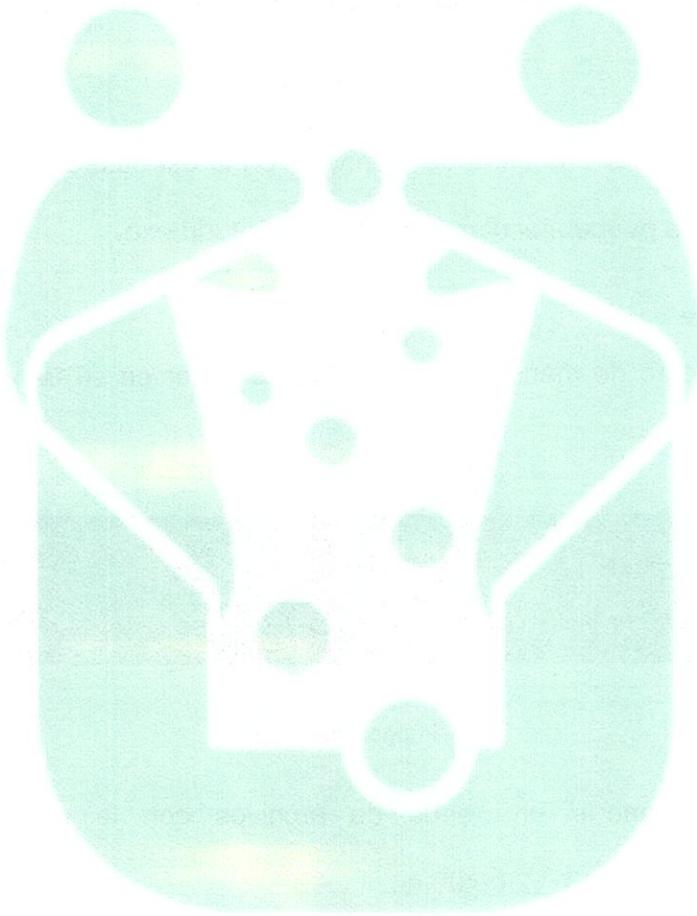
Esto le debe generar una serie de archivos con la estructura similar a met_em.d01.2024-06-13_12:00:00.nc.

Estos archivos con raíz **met_em.d0*** deben ser copiados a la carpeta donde se encuentra el programa ejecutable wrf.exe de la siguiente manera

> cp met_em.d0* ..\WRFV4.5\test\em_real

Estas son las condiciones de frontera que el modelo usará para correr. Hasta aquí se ha finalizado con la configuración y ejecución del módulo WPS. Ahora se debe continuar con el modelo WRF.

NOTA: El asterisco después de una palabra como en el texto anterior (met_em.d0*) significa que la palabra será repetida en todos los archivos, más una serie de caracteres, en este caso fechas, que son diferentes entre un archivo y otro.

 **Danke**
CONSULTORIARFC: DCO120204NC1


3. Modelo WRF-Chem

3.1 Sobre el modelo WRF

El modelo WRF está compuesto por dos programas, el *real.exe* y *wrf.exe* ubicados en el directorio siguiente:

> **/home/wrf1/WorkDir/WRF-Chem/WRFV4.5/test/em_real**

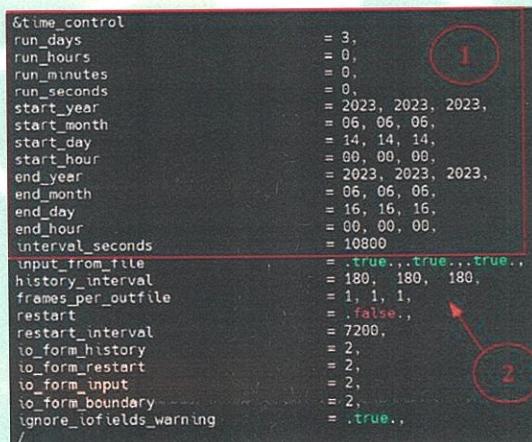
Note que en este directorio deben encontrarse los archivos *met_em.d0* generados en el módulo WPS. Además en este directorio debe encontrar el archivo *namelist.input*, que es donde se ingresan las parametrizaciones y fechas en que se va a correr el modelo.

NOTA: Para correr normalmente el modelo WRF se ejecuta primero el programa *real.exe* y posteriormente *wrf.exe*, sin embargo como en este caso se corre el módulo de química atmosférica se harán pasos extras y se usarán dos archivos *namelist.input* ubicados en el directorio de respaldos en la subcarpeta *templates*.

3.2 Configuración de *namelist.input* para *real.exe*

El archivo *namelist.input* es un archivo de texto donde se indican las fechas, las parametrizaciones dinámicas, físicas y químicas del modelo WRF. A continuación se darán los pasos necesarios para poder configurar un *namelist.input* adecuado para el buen funcionamiento del modelo, antes de ejecutar el programa *real.exe*.

En la primera sección de este archivo se indican las fechas en que se requiere hacer el pronóstico. Estas fechas deben coincidir con las fechas en el archivo namelist.wps y debe tener tantas columnas como dominios, en este ejemplo solo se usarán las primeras dos columnas para el dominio d01 y d02 que se han establecido en el módulo WPS. Ver recuadro rojo en la Figura 6. Además en esta sección también nos es posible indicar otros parámetros, sin embargo se recomienda que se dejen fijos, o al menos que se requieran especificar para otro ejemplo.



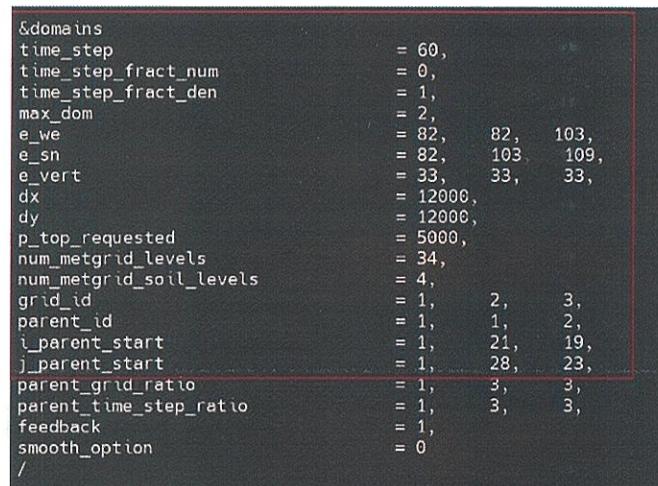
```

&time_control
run_days           = 3,
run_hours          = 0,
run_minutes         = 0,
run_seconds         = 0,
start_year         = 2023, 2023, 2023,
start_month        = 06, 06, 06,
start_day          = 14, 14, 14,
end_year           = 2023, 2023, 2023,
end_month          = 06, 06, 06,
end_day            = 16, 16, 16,
interval_seconds   = 10800
input_from_file    = .true...true...true.,
history_interval   = 180, 180, 180,
frames_per_outfile = 1, 1, 1,
restart            = .false.,
restart_interval   = 7200,
io_form_history    = 2,
io_form_restart    = 2,
io_form_input       = 2,
io_form_boundary   = 2,
ignore_iofields_warning = .true.,
/

```

Figura 6. Ejemplo de sección de control de tiempo del namelist.input. El recuadro rojo indica donde se debe modificar el apartado de tiempo y el número 2 en rojo indica el intervalo en que se guardarán las salidas del modelo. Observe que las fechas corresponden al ejemplo dado en el capítulo anterior.

La siguiente sección corresponde al tamaño de los dominios (Figura 7 recuadro rojo). Debe ingresar los datos exactamente igual que lo hizo en el archivo namelist.wps, de lo contrario un mensaje de error le aparecerá en la pantalla.



```
&domains
  time_step = 60,
  time_step_fract_num = 0,
  time_step_fract_den = 1,
  max_dom = 3,
  e_we = 82, 82, 103,
  e_sn = 82, 103, 109,
  e_vert = 33, 33, 33,
  dx = 12000,
  dy = 12000,
  p_top_requested = 5000,
  num_metgrid_levels = 34,
  num_metgrid_soil_levels = 4,
  grid_id = 1, 2, 3,
  parent_id = 1, 1, 2,
  i_parent_start = 1, 21, 19,
  j_parent_start = 1, 28, 23,
  parent_grid_ratio = 1, 3, 3,
  parent_time_step_ratio = 1, 3, 3,
  feedback = 1,
  smooth_option = 0
/
```

Figura 7. Ejemplo de sección de dominios del namelist.input. El recuadro rojo indica donde se debe modificar el apartado tamaño de dominios.

Además del control de tiempo y tamaño de dominios en este archivo se encontrarán otros apartados, los más importantes son *physics* y *dynamics*. Estas deben ser constantes y no se modifican en cada ocasión que se corra el modelo numérico. Actualmente la configuración correspondiente es la adecuada para el estado de Puebla y regiones cercanas. Esta información se ha obtenido por medio de diversas fuentes bibliográficas y en la página web oficial de buenas prácticas del modelo WRF. Las parametrizaciones físicas más importantes son las mencionadas en la Tabla 1.

Tabla 1. Parametrizaciones físicas recomendadas del modelo WRF para un dominio en el centro de la República Mexicana. Las parametrizaciones aquí indicadas son las que actualmente se encuentran operativas en el modelo WRF Chem.

| variable | descripción | opción numérica |
|----------------|---------------------------------------|-----------------|
| mp_physics | WSM 6-class graupel scheme | 6 |
| cu_physics | Kain-Fritsch scheme | 1 |
| ra_lw_physics | Rapid Radiative Transfer Model scheme | 1 |
| ra_sw_physics | Dudhia scheme | 1 |
| bl_pbl_physics | YSU scheme | 1 |

Si bien podrá observar otras opciones en las secciones mencionadas, es recomendado no modificar estos parámetros, ya que puede llevar al mal funcionamiento del modelo WRF. Un ejemplo de namelist.input puede ser visto en el Apéndice A.

Condiciones iniciales y de frontera

Puesto que se cuenta con una configuración adecuada, ahora se debe correr el archivo real.exe que se encuentra en el directorio de trabajo. Para ello es recomendable usar la siguiente instrucción en su terminal:

> mpirun -np 12 ./real.exe

Si el namelist.input se ha llenado adecuadamente al correr este programa le deberá generar las condiciones de frontera del modelo, así como mensaje de tarea exitosa al final. El nombre de los archivos que debe generar son wrfbdy_d0*, y wrfinput_d0*, éste último debe generarse uno para cada dominio. En este ejemplo son dos. Es preciso seguir este proceso, ya que los archivos de condiciones de frontera serán la base principal para generar los archivos de contaminantes que leerá el modelo WRF.

3.3 Integración de emisiones al modelo WRF

El modelo WRF es un modelo numérico para investigación y pronóstico meteorológico. El módulo que genera los procesos químicos debe iniciar con datos de emisiones los cuales deben cumplir con un formato estricto, por lo tanto, se usan las condiciones de frontera generadas por el programa real.exe, esto es, los archivos wrfinput_d01 y wrfinput_d02 que se encuentra en el directorio de trabajo, que servirán como plantilla para poder ingresar las emisiones.

Para generar una base de datos que cumpliera con la estructura requerida por el modelo WRF se desarrollaron los siguientes programas que están ubicadas en el directorio ~/home/wrf1/Workdir/scripts y que a continuación se describen.

addEmissions.py

Este programa es una función desarrollada en lenguaje python y que fue diseñado para que lea los archivos (extensión *.lsi*) que contienen las observaciones *in situ*, las interpola a una malla regular replicada de los archivos geo_em.d0*.nc originados por geogrid.exe en el módulo WPS y los guarda en formato netcdf con sus respectivos metadatos. Las emisiones son leídas del directorio ~ /home/wrf1/Workdir/emisiones y captura las últimas 12 horas de datos (puede ser editable en el código, instrucciones en el código) y rellena huecos en caso de haberlos. Esta función es usada por el programa wrfchemi_add_emissions.py que se encuentra en el mismo directorio y que genera los archivos **emis_d0***.

wrfchemi_zeros.R

Este programa está elaborado en lenguaje R script, está diseñado para que lea los archivos **wrfbdy_d0*** que se encuentran en el directorio *em_real* y que fueron generados por real.exe. Este programa copia la estructura de metadatos y datos que requiere el modelo WRF-Chem de los archivos de condiciones de frontera e iniciales wrfbdy y genera archivos en formato netcdf (solo serán temporales) que guarda en la carpeta scripts con la raíz **emis_d0***. En el código del programa se encuentran detalladas las instrucciones si desea cambiar el número de dominios, para este ejercicio está fijo en dos.

addEmissions.R

No confundir con addEmissions.py. Este programa está elaborado en lenguaje R script, está diseñado para que lea los archivos **wrfchemi*** generados por el programa wrfchemi_zeros.R e integra las emisiones guardadas en los archivos **emis_d0***, obteniendo una combinación de ambos. Para generar emisiones en las ubicaciones de las estaciones, estos datos permanecerán constantes (emitiendo) durante el periodo de pronóstico, y se actualizará cada ocasión que se ejecute todo el proceso de pronóstico autónomo (explicado en las siguientes secciones). Estos archivos se envían a la carpeta *em_real* o se ligan para que se encuentren disponibles para el modelo wrf.exe que se encuentra en el mismo directorio.

wrfchemi_zeros.sh

No confundir con wrfchemi_zeros.R. Este programa está elaborado en lenguaje bash script y fue elaborado con el objetivo de optimizar el proceso de elaboración de condiciones iniciales químicas para el modelo WRF. Este programa está encargado de ejecutar paso a paso y en el orden correcto los programas descritos anteriormente. Este programa también cambia el nombre de los archivos al formato correcto que es **wrfchemi_d0*_yyyy-mm-dd hh:00:00** y los mueve a la carpeta *em_real*. Si se requieren generar estos archivos, solo es necesario ejecutar este programa en su terminal con la instrucción bash.

El orden de ejecución del proceso sería el siguiente

- > **python wrfchemi_add_emissions.py**
- > **Rscript wrfchemi_zeros.R**
- > **Rscript addEmissions.R**

Posteriormente, debe mover los archivos wrfchemi* generados en el directorio *script* al directorio *em_real* y renombrarlos con el formato wrfchemi_d0*_yyyy-mm-dd_hh:00:00.

Sin embargo, para evitar los pasos anteriores solo debe ejecutar la siguiente instrucción en su terminal

> **bash wrfchemi_zeros.sh**

Como se describió anteriormente, este programa ejecuta paso a paso el proceso de elaboración de condiciones iniciales de química atmosférica. Este programa también moverá los archivos al directorio *em_real*.

3.4 Configuración de namelist.input para wrf.exe

Ya que se han generado las condiciones iniciales y de frontera por el programa real.exe (sección 3.2) y las condiciones iniciales de química atmosférica generadas por el programa wrfchemi_zeros.sh (sección 3.3) debe configurar adecuadamente el archivo namelist.input para poder ejecutar el programa wrf.exe y que el modelo pueda leer las emisiones de contaminantes en los archivos wrfchemi_d0*. Para esto, el archivo namelist.input debe ser modificado en la sección *time_control* agregando las siguientes líneas que están ilustradas en la figura 8.

```

# nombre del archivo wrfchemi_d*
auxinput5_inname = 'wrfchemi_d<domain>_<date>'

# tiempo en minutos, intervalo de entrada de datos de emisiones
auxinput5_interval_m = 360, 360,

# número de frames por archivo de entrada. En este ejercicio son 13
frames_per_auxinput5 = 13, 13,

```

Formato de entrada en NetCDF opción 2.

```

io_form_auxinput5 = 2,

```

| | |
|----------------------|-------------------------------|
| auxinput5_inname | = 'wrfchemi_d<domain>_<date>' |
| auxinput5_interval_m | = 360, 360, 360, |
| frames_per_auxinput5 | = 13, 13, 13, |
| restart | = .false., |
| restart_interval | = 7200, |
| io_form_history | = 2, |
| io_form_restart | = 2, |
| io_form_input | = 2, |
| io_form_boundary | = 2, |
| io_form_auxinput5 | = 2, |

Figura 8. Líneas agregadas en el namelist.input en la sección de control de tiempo.

Además debe agregar las líneas mencionadas inmediato al apartado &bdy_control (ver Apéndice A) debe agregar un apartado específico para las parametrizaciones de química atmosférica como en la figura 9, en este caso se sugiere usar la plantilla que se encuentra en el directorio respaldos o la agregada en la sección de Apéndices de este manual. Ésta configuración ha sido probada y es compatible con el formato de datos de entrada de química atmosférica. Si requiere conocer más sobre otro tipo de configuraciones, se recomienda acudir al manual de usuario

oficial. Cambiar alguno de los parámetros (opciones) en alguna de las columnas en el archivo namelist.input recomendado al usuario puede llevárselo a generar el mal funcionamiento del modelo.

```

&chem
kemit
chem_opt
bioemdt
photdt
chemdt
io_style_emissions
emiss_inpt_opt
emiss_opt
chem_in_opt
phot_opt
gas_drydep_opt
aer_drydep_opt
bio_emiss_opt
dust_opt
dmsemis_opt
seas_opt
gas_bc_opt
gas_ic_opt
aer_bc_opt
aer_ic_opt
gaschem_onoff
aerchem_onoff
wetsav_onoff
cldchem_onoff
vertmix_onoff
chem_conv_tr
biomass_burn_opt
plumerisefire_frg
aer_ra_feedback
have_bcs_chem
/
= 1,
= 1,    1,    1,
= 30,   30,   30,
= 30,   30,   30,
= 0.,   0.,   0.,
= 2,
= 1,    1,    1,
= 3,    3,    3,
= 1,    1,    1,
= 1,    1,    1,
= 1,    1,    1,
= 1,    1,    1,
= 0,
= 0,
= 0,
= 1,    1,    1,
= 1,    1,    1,
= 1,    1,    1,
= 1,    1,    1,
= 1,    1,    1,
= 0,    0,    0,
= 0,    0,    0,
= 1,    1,    1,
= 1,    1,    1,
= 1,    1,    1,
= 0,    0,    0,
= 30,   30,   30,
= 0,    0,    0,
=.false., .false., .false.,
/

```

Figura 9. Líneas agregadas en el namelist.input para módulo de química atmosférica.

3.4.1 Pronóstico con WRF-Chem

Ya que se cuenta con una configuración adecuada, ahora se debe correr el archivo wrf.exe que se encuentra en el directorio de trabajo. Para ello es recomendable usar la siguiente instrucción en su terminal:

> mpirun -np 30 ./wrf.exe

Observe que a diferencia con el real.exe en este caso la opción 12 ha cambiado a 30, lo cual significa que el proceso de pronóstico se estará elaborando con mayor cantidad de recursos de cómputo de su servidor. Si el ramelist.input se ha llenado

adecuadamente al correr este programa le deberá generar las salidas en formato

netcdf con el siguiente formato

wrfout_d0*_yyyy-mm-dd_hh:00:00

Cuando termine de ejecutarse todo el proceso deberá aparecer un mensaje de proceso exitoso. Si requiere ver el proceso puede realizar la siguiente acción:

> tail -f rsl.error.0000

Y le desplegará la información en su terminal

4. Sistema autónomo para pronóstico meteorológico

4.1 Elaboración autónoma de pronóstico

Los pasos descritos en secciones anteriores se resumieron en un solo programa que elabora paso a paso todo el proceso de pronóstico, desde la descarga de condiciones de frontera del modelo GFS hasta correr el modelo WRF. Este programa fue elaborado en código bash script y será el encargado de ejecutar el modelo WRF tantas veces sea necesario sin requerir de modificaciones y evitando la intervención del usuario para evitar errores en el proceso.

Para el funcionamiento de este programa se requieren de plantillas con las parametrizaciones correctas de los namelist usados en secciones anteriores. Para ello se cuenta con un respaldo en el directorio ~wrf1/respaldos/templates donde el programa autónomo copiará la base y hará las modificaciones necesarias cada ocasión que el programa se ejecute.

4.1.1 Iniciar el sistema desde la terminal

Para iniciar el proceso de pronóstico debe ingresar al directorio de ejecución del modelo en el directorio *model-run*.

> cd /home/wrf1/Workdir/model-run

En este directorio podrá encontrar el programa llamado model-run.sh (puede usar ls en la terminal para poder verlo), el cual es el encargado de realizar todo el proceso de pronóstico. Para ejecutarlo una vez posicionado en la carpeta correspondiente puede hacerlo de la siguiente manera.

> bash model-run.sh

Una vez ejecutado el proceso el programa iniciará con la descarga de datos y posteriormente con los pasos descritos en este manual.

4.1.2 Tarea en crontab

Para que se genere el pronóstico de manera automática en el servidor del usuario se puede programar una tarea a la hora deseada. Por ejemplo, si se requiere que el pronóstico se realice a las 12:30hrs solo se requiere programar la tarea. Para el uso de crontab puede teclear en su terminal

> crontab -e

Inmediatamente le aparecerá un editor de texto donde puede colocar la siguiente instrucción.

30 12 * * * bash /home/wrf1/WorkDir/model-run/model-run.sh > model.log 2>&1

Lo anterior descrito significa que el modelo será ejecutado a las 12 horas del día con 30 minutos. Para fines operativos en el servidor se ha fijado la hora a las 03:30hrs considerando que es un tiempo adecuado para descargar las condiciones

de frontera de las 00:00Z del modelo GFS. Si desea que el sistema no genere el pronóstico sólo debe comentar la línea señalada anteriormente.

En el ejemplo anterior se observa el comentario al final de la instrucción.

4.2 Problemas conocidos

Este tipo de sistemas de pronóstico meteorológico pueden quedar detenidos arbitrariamente, por lo que es común encontrar errores en la ejecución del sistema.

Es por eso que se listan los siguientes problemas conocidos.

- Los datos de condiciones de frontera no se descargaron. Los datos del modelo GFS no se descargaron correctamente, para ésta acción vaya al directorio de model-run y vuelva a correr el programa gfsrun.sh o todo el sistema completo con el programa model-run-sh en su terminal como se ha indicado en la sección 1.3 o 4.1.1. Para verificar que se hayan descargado correctamente los datos verifique en el directorio de descarga *gfs*. En casos muy aislados la dirección web de descarga de datos del modelo GFS es modificada por el propietario, para esto ingrese en el script gfsrun.sh y modifique correctamente el sitio web. Es importante mencionar que esta última acción es muy difícil que suceda.
- El pronóstico no se generó, no se corrió correctamente el módulo WPS o el programa se quedó estancado y no generó el pronóstico. En estos caso es recomendable ir a la carpeta model-run y ejecutar el programa con el mismo nombre como en la sección 4.1.1, en algunas ocasiones el sistema no inicia correctamente, por lo que deberá iniciar de nuevo el programa. Para evitar que se descarguen del modelo GFS de nuevo puede ingresar al programa

model-run.sh y comentar la línea de descarga de datos #bash gfsrun.sh para evitar el tiempo de descarga de datos. Es importante que una vez que ejecute todo el sistema regrese el programa model-run.sh a su configuración habitual. Verifique que el modelo corrió correctamente en el directorio *Workdir/wrf_out*.

- Se generó el pronóstico pero no se generaron las gráficas. Es común que en sistemas como este algunos procesos se queden estancados, y uno de ellos es la generación de gráficas. En este caso no es necesario volver a correr todo el sistema de pronóstico. Solo debe ingresar al directorio scripts verificar si en realidad no se generaron las gráficas en el subdirectorio *runs* y si así es solo debe ingresar en su terminal la siguiente instrucción.

> cd /home/wrf1/Workdir/scripts

> bash runplots.sh

El programa runplots.sh es un programa elaborado en bash script que ejecuta uno a uno los programas que están encargados de generar las gráficas de las salidas del modelo WRF. En caso de querer solo una variable en particular solo debe ingresar en la terminal la variable y/o el nivel deseado, por ejemplo, si requiere el viento en el nivel de 500mb solo tiene que ejecutar en la terminal

> python plot_3hr_wnd500.py

- No se generaron los archivos wrfchemi_*. Solo debe ingresar al directorio scripts verificar si en realidad no se generaron los archivos y ejecute el programa wrfchemi_zeros.sh.

> cd /home/wrf1/Workdir/scripts

> bash wrfchemi_zeros.sh

- Cambió el link de descarga de datos del modelo GFS. En casos muy aislados la dirección web de descarga de datos del modelo GFS es modificada por el propietario, para esto ingrese en el script gfsrun.sh y modifique correctamente el sitio web indicado en el mismo programa. Es importante mencionar que este último problema es muy difícil que suceda.

Apéndice A

Ejemplo de namelist.wps configurado para el estado de Puebla con dos dominios.

&share

```
wrf_core = 'ARW',
max_dom = 2,
start_date = '2024-06-13_12:00:00', '2024-06-14_12:00:00',
end_date = '2024-06-16_12:00:00', '2024-06-16_12:00:00',
interval_seconds = 10800,
io_form_geogrid = 2,
```

```
/
```

&geogrid

```
parent_id = 1, 1,
parent_grid_ratio = 1, 3,
i_parent_start = 1, 21,
j_parent_start = 1, 28,
e_we = 82, 82,
e_sn = 82, 103,
geog_data_res = '30s','30s',
dx = 12000,
dy = 12000,
map_proj = 'mercator',
ref_lat = 19.00,
ref_lon = -97.00,
truelat1 = 30.0,
truelat2 = 60.0,
stand_lon = -79.0,
geog_data_path = '/home/wrf1/WorkDir/WRF-Chem/geog/WPS_GEOG/'
```

&ungrid

```
out_format = 'WPS',
prefix = 'FILE',
/
```

&metgrid

```
fg_name = 'FILE',
io_form_metgrid = 2,
/
```

Ejemplo de namelist.input configurado para el estado de Puebla con dos dominios (puede omitir la tercera columna). Este namelist.input servirá para generar las condiciones de frontera.

&time_control

```
run_days           = 3,
run_hours          = 0,
run_minutes         = 0,
run_seconds         = 0,
start_year          = 2023, 2023, 2023,
start_month         = 06, 06, 06,
start_day           = 14, 14, 14,
start_hour          = 00, 00, 00,
end_year            = 2023, 2023, 2023,
end_month           = 06, 06, 06,
end_day             = 16, 16, 16,
end_hour            = 00, 00, 00,
interval_seconds    = 10800
input_from_file     = .true.,.true.,.true.,
history_interval   = 180, 180, 180,
```

```

frames_per_outfile      = 1, 1, 1,
restart                 = .false.,
restart_interval         = 7200,
io_form_history          = 2,
io_form_restart           = 2,
io_form_input              = 2,
io_form_boundary           = 2,
ignore_iofields_warning   = .true.,
/

```

&domains

```

time_step                = 60,
time_step_fract_num       = 0,
time_step_fract_den        = 1,
max_dom                  = 2,
e_we                      = 82, 82, 103,
e_sn                      = 82, 103, 109,
e_vert                     = 33, 33, 33,
dx                         = 12000,
dy                         = 12000,
p_top_requested            = 5000,
num_metgrid_levels          = 34,
num_metgrid_soil_levels     = 4,
grid_id                    = 1, 2, 3,
parent_id                  = 1, 1, 2,
i_parent_start              = 1, 21, 19,
j_parent_start              = 1, 28, 23,
parent_grid_ratio             = 1, 3, 3,
parent_time_step_ratio       = 1, 3, 3,
feedback                   = 1,

```

smooth_option = 0

/

&physics

physics_suite = 'CONUS'
 mp_physics = 6, 6, 6,
 cu_physics = 1, 1, 1,
 ra_lw_physics = 1, 1, 1,
 ra_sw_physics = 1, 1, 1,
 bl_pbl_physics = 1, 1, 1,
 sf_sfclay_physics = 1, 1, 1,
 sf_surface_physics = 2, 2, 2,
 radt = 3, 3, 3,
 bldt = 0, 0, 0,
 cudt = 5, 5, 5,
 icloud = 1,
 num_land_cat = 21,
 sf_urban_physics = 0, 0, 0,

/

&fdda

/

&dynamics

hybrid_opt = 2,
 w_damping = 0,
 diff_opt = 1, 1, 1,
 km_opt = 4, 4, 4,
 diff_6th_opt = 0, 0, 0,
 diff_6th_factor = 0.12, 0.12, 0.12,
 base_temp = 290.,
 damp_opt = 3,

```

zdamp          = 5000., 5000., 5000.,
dampcoef       = 0.2,   0.2,   0.2
khdif          = 0,     0,     0,
kvdif          = 0,     0,     0,
non_hydrostatic = .true., .true., .true.,
moist_adv_opt  = 1,     1,     1,
scalar_adv_opt = 1,     1,     1,
gwd_opt         = 1,
/
&bdy_control
spec_bdy_width = 5,
specified        = .true.
/
&grib2
/
&namelist_quilt
nio_tasks_per_group = 0,
nio_groups = 1,
/

```

Ejemplo de namelist.input configurado para el estado de Puebla con dos dominios y con parametrizaciones de química atmosférica.

```

&time_control
run_days        = 3,
run_hours        = 0,
run_minutes      = 0,
run_seconds      = 0,
start_year       = 2023, 2023, 2023,
start_month      = 06, 06, 06,
start_day        = 14, 14, 14,

```

```

start_hour          = 00, 00, 00,
end_year           = 2023, 2023, 2023,
end_month          = 06, 06, 06,
end_day            = 16, 16, 16,
end_hour           = 00, 00, 00,
interval_seconds   = 10800
input_from_file    = .true.,.true.,.true.,
history_interval   = 180, 180, 180,
frames_per_outfile = 1, 1, 1,
auxinput5_inname   = 'wrfchemi_d<domain>_<date>'
auxinput5_interval_m = 360, 360, 360,
frames_per_auxinput5 = 13, 13, 13,
restart            = .false.,
restart_interval   = 7200,
io_form_history    = 2,
io_form_restart    = 2,
io_form_input      = 2,
io_form_boundary   = 2,
io_form_auxinput5  = 2,
ignore_iofields_warning = .true.,
/

```

&domains

```

time_step           = 60,
time_step_fract_num = 0,
time_step_fract_den = 1,
max_dom            = 2,
e_we               = 82, 82, 103,
e_sn               = 82, 103, 109,
e_vert              = 33, 33, 33,

```



```

dx          = 12000,
dy          = 12000,
p_top_requested      = 5000,
num_metgrid_levels   = 34,
num_metgrid_soil_levels = 4,
grid_id        = 1, 2, 3,
parent_id       = 1, 1, 2,
i_parent_start   = 1, 21, 19,
j_parent_start   = 1, 28, 23,
parent_grid_ratio = 1, 3, 3,
parent_time_step_ratio = 1, 3, 3,
feedback         = 1,
smooth_option    = 0
/

```

&physics

```

physics_suite      = 'CONUS'
mp_physics        = 6, 6, 6,
cu_physics        = 5, 5, 5,
ra_lw_physics     = 1, 1, 1,
ra_sw_physics     = 1, 1, 1,
bl_pbl_physics    = 1, 1, 1,
sf_sfclay_physics = 1, 1, 1,
sf_surface_physics= 2, 2, 2,
radt              = 3, 3, 3,
bldt              = 0, 0, 0,
cudt              = 5, 5, 5,
icloud             = 1,
num_land_cat      = 21,
sf_urban_physics  = 0, 0, 0,

```

```

cu_diag           = 1,   1,
cu_rad_feedback      = .true.,
/
&fdda
/
&dynamics
hybrid_opt        = 2,
w_damping          = 0,
diff_opt           = 1,   1,   1,
km_opt             = 4,   4,   4,
diff_6th_opt       = 0,   0,   0,
diff_6th_factor    = 0.12, 0.12, 0.12,
base_temp          = 290.
damp_opt           = 3,
zdamp              = 5000., 5000., 5000.,
dampcoef           = 0.2, 0.2, 0.2
khdif              = 0,   0,   0,
kvdif              = 0,   0,   0,
non_hydrostatic     = .true., .true., .true.,
moist_adv_opt      = 1,   1,   1,
scalar_adv_opt     = 1,   1,   1,
chem_adv_opt       = 1,   1,   1,
gwd_opt             = 1,
/
&bdy_control
spec_bdy_width     = 5,
specified           = .true.
/

```

&chem

| | |
|--------------------|---------------|
| kemit | = 1, |
| chem_opt | = 1, 1, 1, |
| bioemdt | = 30, 30, 30, |
| photdt | = 30, 30, 30, |
| chemdt | = 0., 0., 0., |
| io_style_emissions | = 2, |
| emiss_inpt_opt | = 1, 1, 1, |
| emiss_opt | = 3, 3, 3, |
| chem_in_opt | = 1, 1, 1, |
| phot_opt | = 1, 1, 1, |
| gas_drydep_opt | = 1, 1, 1, |
| aer_drydep_opt | = 1, 1, 1, |
| bio_emiss_opt | = 1, 1, 1, |
| dust_opt | = 0, |
| dmsemis_opt | = 0, |
| seas_opt | = 0, |
| gas_bc_opt | = 1, 1, 1, |
| gas_ic_opt | = 1, 1, 1, |
| aer_bc_opt | = 1, 1, 1, |
| aer_ic_opt | = 1, 1, 1, |
| gaschem_onoff | = 1, 1, 1, |
| aerchem_onoff | = 1, 1, 1, |
| wetscav_onoff | = 0, 0, 0, |
| cldchem_onoff | = 0, 0, 0, |
| vertmix_onoff | = 1, 1, 1, |
| chem_conv_tr | = 1, 1, 1, |
| biomass_burn_opt | = 0, 0, 0, |
| plumerisfire_frq | = 30, 30, 30, |



Calle: Tehuacán Sur, 69, Col: La Paz, CP: 72160, Puebla, Puebla, México

Tel. 2224676424 RFC: DCO120204NC1

www.dankeconsultoria.com.mx

aer_ra_feedback = 0, 0, 0,

have_bcs_chem = .false., .false., .false.,

/

&grib2

/

&namelist_quilt

nio_tasks_per_group = 0,

nio_groups = 1,

/