PERÚ Ministerio del Ambiente

Senamhi
SERVICIO NACIONAL DE METEOROLOGÍA E HIDROLOGÍA DEL PERÚ

Siempre con el pueblo

Taller de modelamiento climático

Instructores:

Gerardo Jacome Vergaray

Jorge Llamocca Huamaní

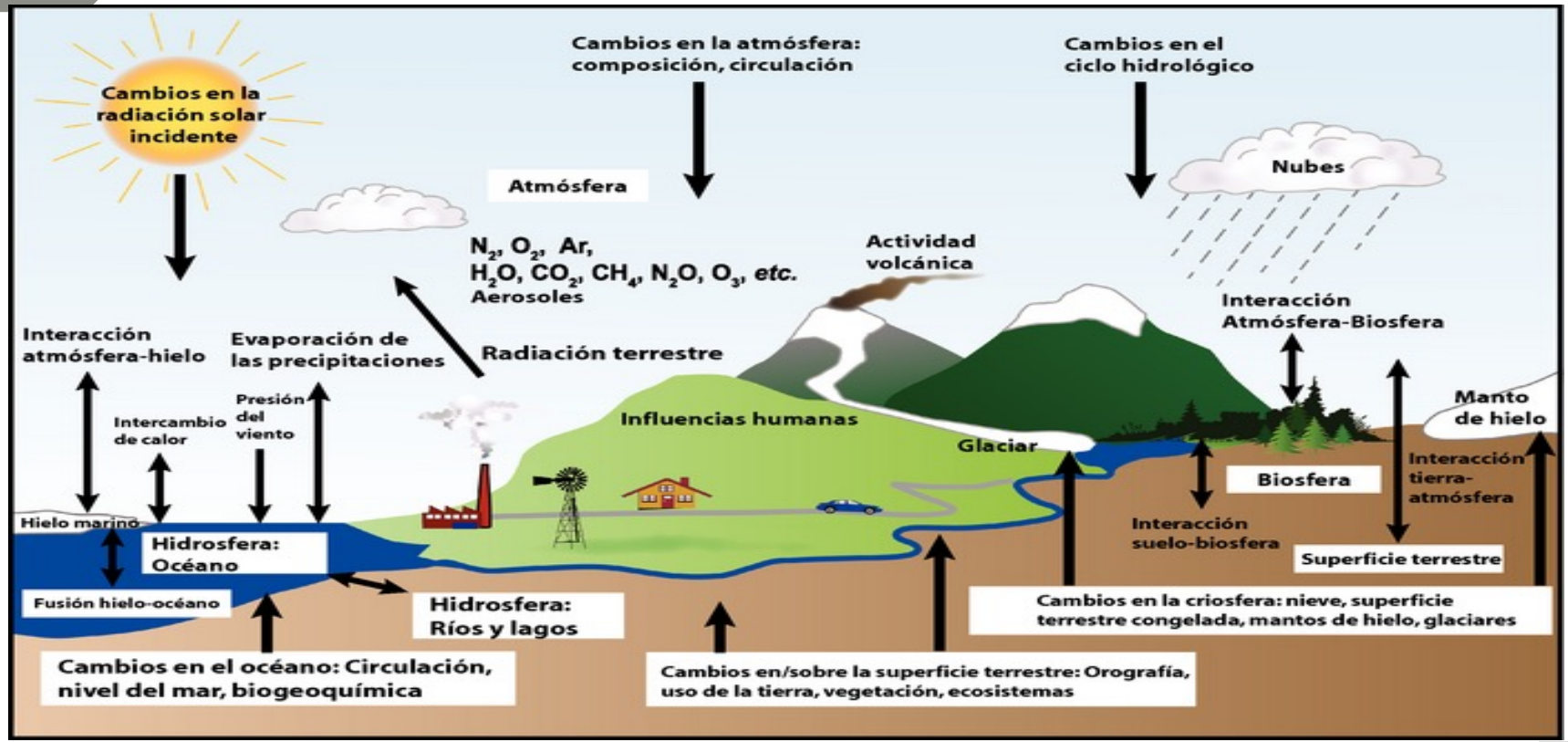
Subdirección de modelamiento Numérico de la Atmosfera

Dirección de Meteorología y Evaluación Ambiental Atmosférica

Contenido

- SISTEMA CLIMÁTICO
- MODELO CLIMÁTICO GLOBAL
- MODELO CLIMÁTICO REGIONAL
- MODELO REGIONAL WRF
- RECURSO COMPUTACIONAL HPC
- COMPILACIÓN DEL MODELO WRF
- PRACTICA: ACCESO AL SERVIDOR DE SENAMHI

SISTEMA CLIMÁTICO



Atmósfera (aire), hidrósfera (agua), criósfera (hielo y permafrost), la litósfera (suelo) y la Biósfera (seres vivos)

MODELO CLIMÁTICO GLOBAL

$$\frac{\partial \overline{V}}{\partial t} + \overline{V} \cdot \nabla \overline{V} = -\frac{\nabla p}{\rho} - 2\overline{\Omega} \times \overline{V} + \overline{g} + \overline{F_V}$$

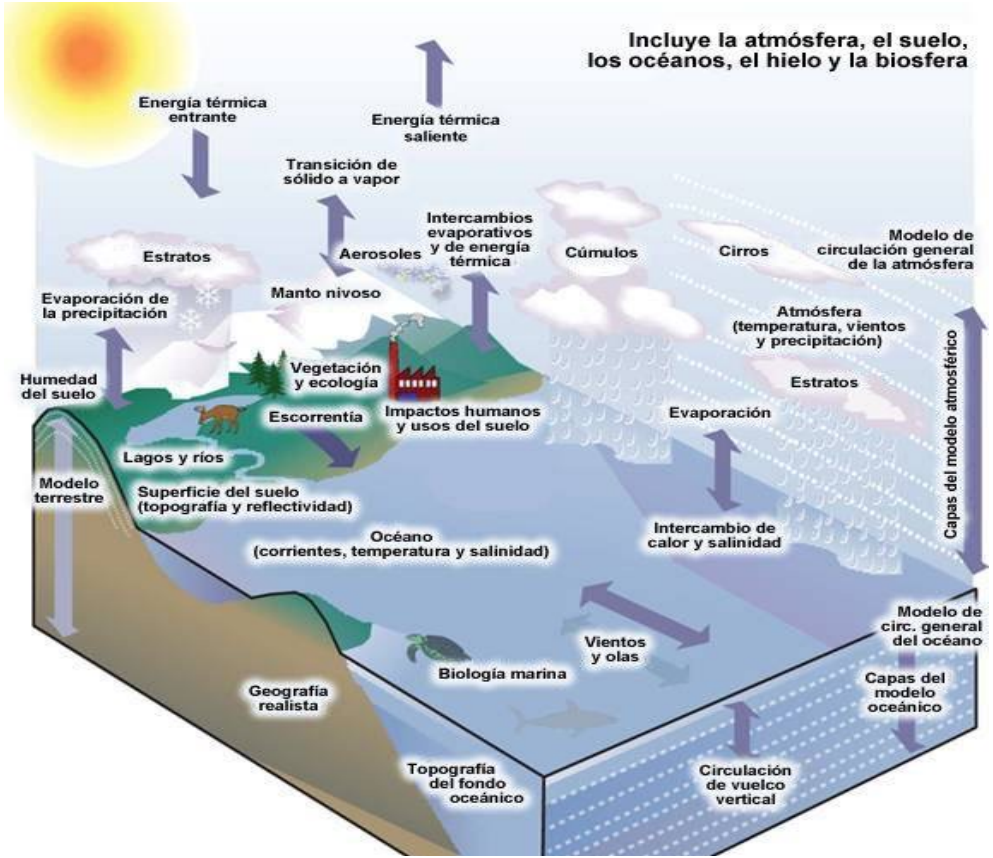
$$C_p\left(\frac{\partial T}{\partial t} + \overline{V} \cdot \nabla T\right) = \frac{1}{\rho}\frac{dp}{dt} + Q + F_T$$

$$\frac{\partial \rho}{\partial t} + \overline{V} \cdot \nabla \rho = -\rho \nabla \cdot \overline{V}$$

$$\frac{\partial q}{\partial t} + \overline{V} \cdot \nabla q = \frac{S_q}{\rho} + F_q$$

$$p = \rho R T$$

Son representaciones matemáticas del sistema climático de la tierra, se basan en las leyes de la física, dinámica de fluidos, la química, entre otros.

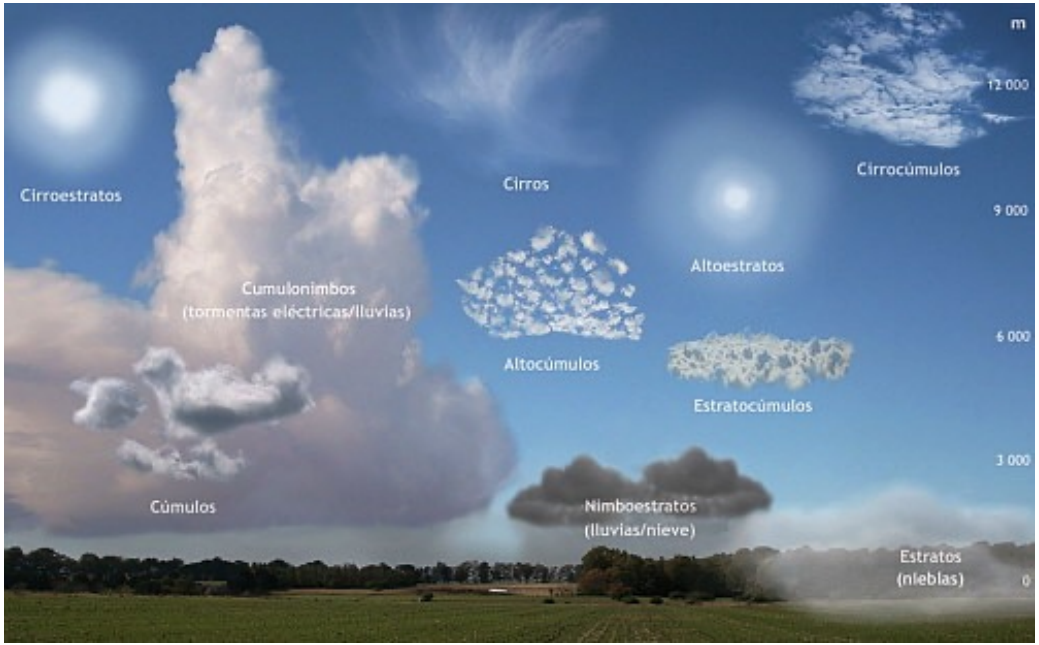


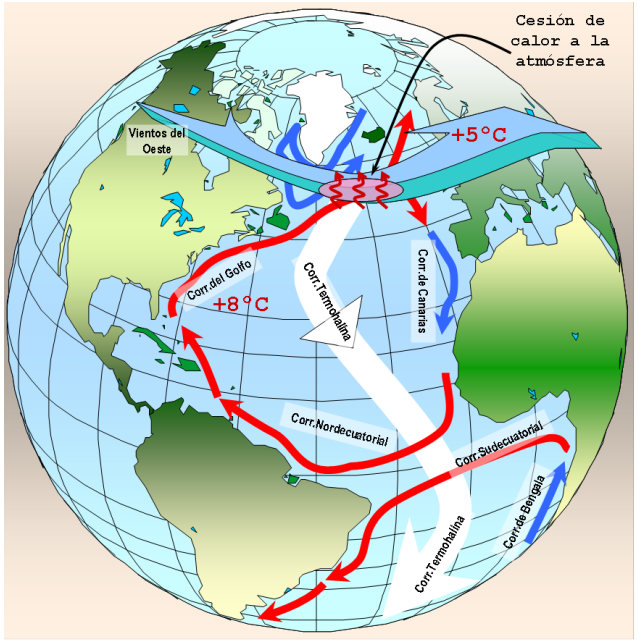
MODELO CLIMÁTICO GLOBAL

Representan como la radiación solar en transmitida, reflejada y absorbida dentro de la atmósfera y en la superficie terrestre

MODELO CLIMÁTICO GLOBAL

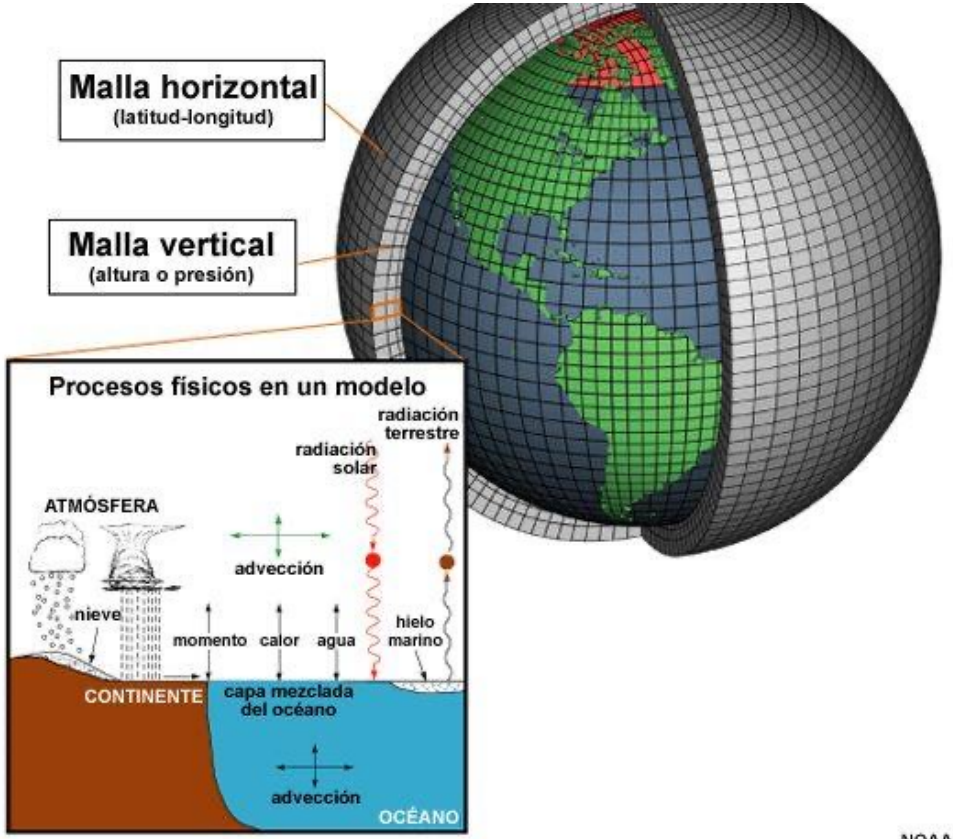
Representan los vientos, las nubes, la lluvia, la circulación de los océanos, etc.





MODELO CLIMÁTICO GLOBAL

Los modelos climáticos globales dividen al planeta en miles de rejillas o mallas tridimensionales, a cada rejilla se le asigna valores únicos de las variables atmosféricas como, el viento, la temperatura, presión, humedad, entre otros.



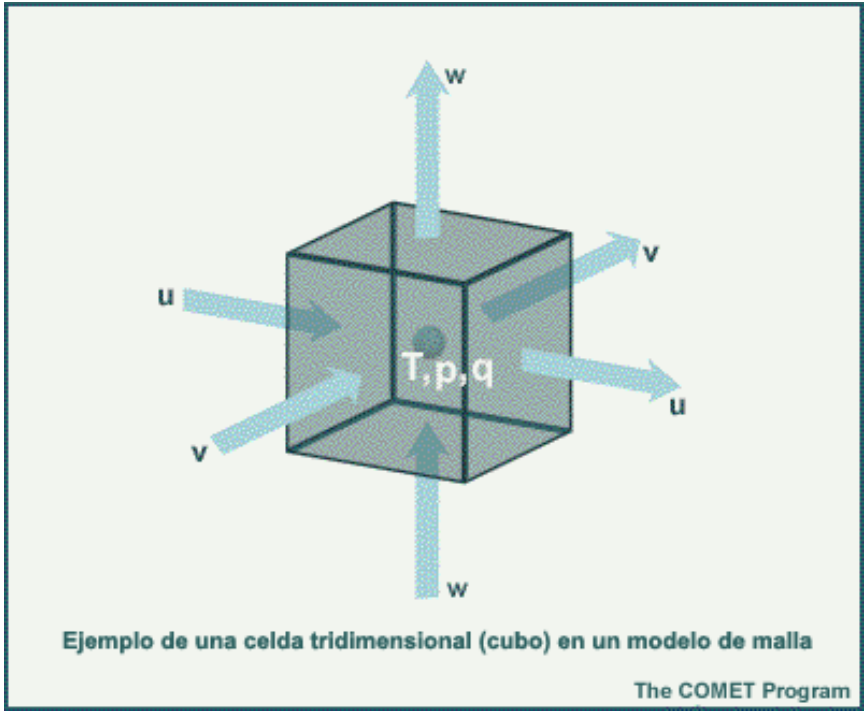
NOAA

REPRESENTACIÓN DE LOS DATOS: PUNTO DE MALLA

Los modelos de malla representan la atmósfera en cubos tridimensionales.

La temperatura, la presión y la humedad (T, p, q) que se indican en el centro del cubo representan el promedio de las condiciones en el interior del cubo.

De forma análoga, los vientos de este a oeste (u) y de norte a sur (v)



Ejemplo de una celda tridimensional (cubo) en un modelo de malla

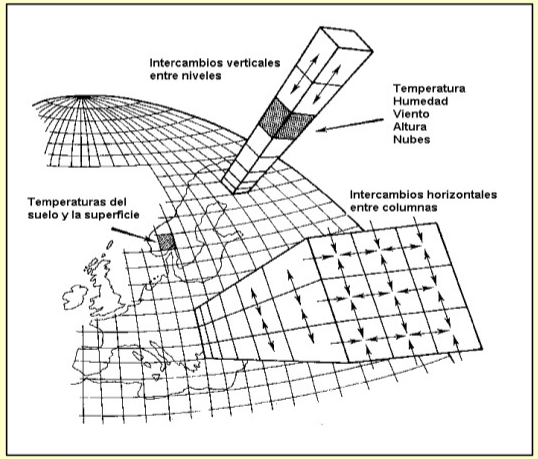
The COMET Program

REPRESENTACIÓN DEL SISTEMA CLIMÁTICO

Aproximaciones numéricas:

$$\frac{\partial V}{\partial t} = m\frac{\partial V}{\partial s} \Rightarrow \Delta V \cong \Delta t \cdot m\frac{\Delta V}{\Delta s} \Rightarrow V(t+\Delta t) = V(t) + \Delta V$$

Discretizando la variación espacial en celdillas 3D(Δx, Δy, Δp)



~ 300 × 300 km atmosf.
~ 100 × 100 km océano

Discretizando la evolución temporal en intervalos finitos (Δt)

Valores de variables en la celdilla (i, j, k)
Viento (u, v)
Temperatura (T)
Presión (p)
Humedad (q)
. . .
en el inicio de la simulación (t_0)

→

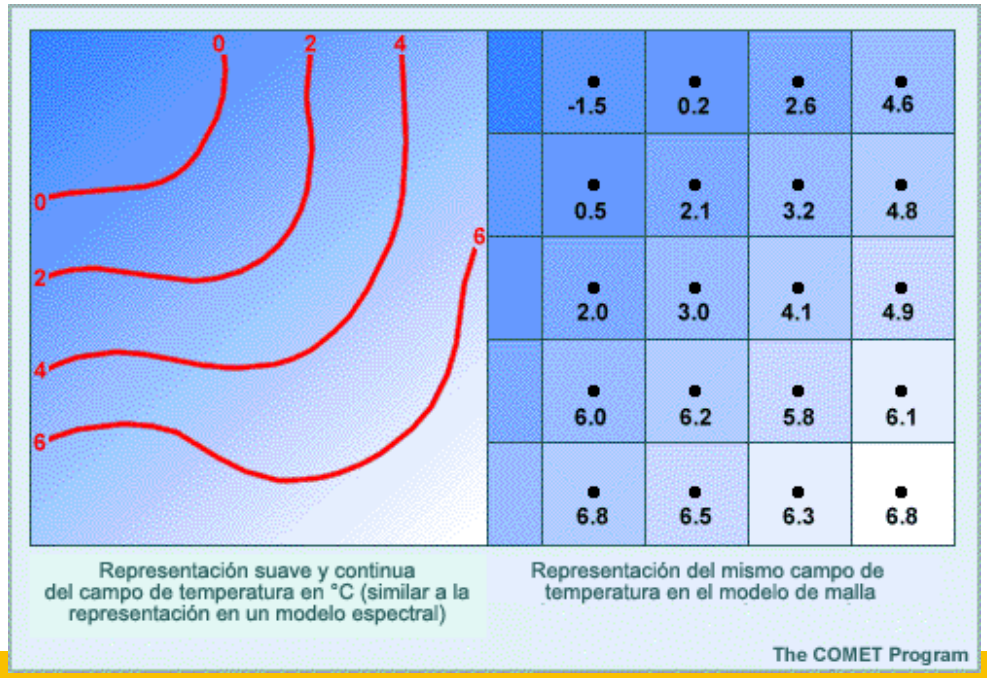
Valores de variables en la celdilla (i, j, k)
Viento (u, v)
Temperatura (T)
Presión (p)
Humedad (q)
. . .
en un instante posterior ($t_0+\Delta t$)

→

... y así se continua avanzando en pasos de tiempo Δt hasta completar el periodo de tiempo de simulación (≈ 250 años)

REPRESENTACIÓN DE LOS DATOS: PUNTO DE MALLA

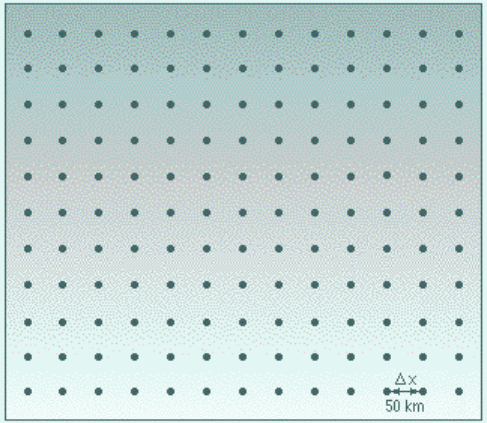
Para resolver las ecuaciones de pronóstico, la mayoría de los modelos numéricos utilizan datos representados como valores sobre una malla. Esta es una representación de puntos de malla.



REPRESENTACIÓN DE LOS DATOS: RESOLUCIÓN HORIZONTAL

La resolución horizontal de un modelo numéricos está relacionada con la distancia entre los puntos de malla o rejilla.

Se define como la distancia media entre puntos de malla adyacentes que utilizan las mismas variables.
En el ejemplo, todas las variables se calculan en cada punto de la malla, de modo que la resolución es 50 km.
Así mismo un modelo similar con una distancia de 10 km entre puntos de malla adyacentes tendría una resolución horizontal de 10 km.



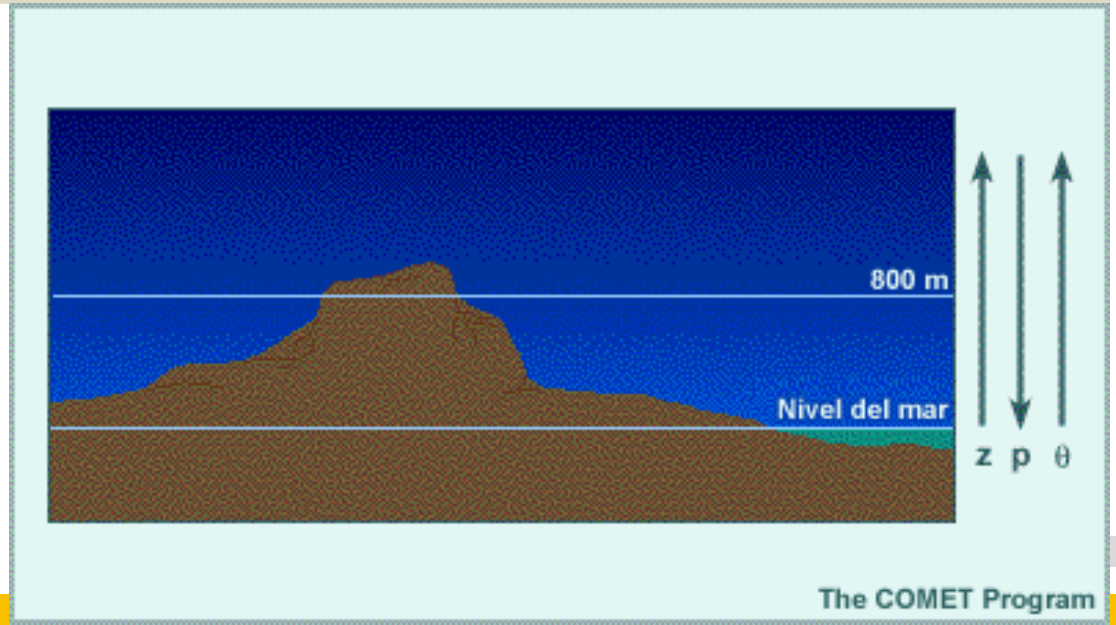
Ejemplo de un modelo de predicción numérica de resolución de 50 km (todas las variables se calculan en cada punto de malla)

The COMET Program

REPRESENTACIÓN DE LOS DATOS: COORDENADA VERTICAL – PRESIÓN

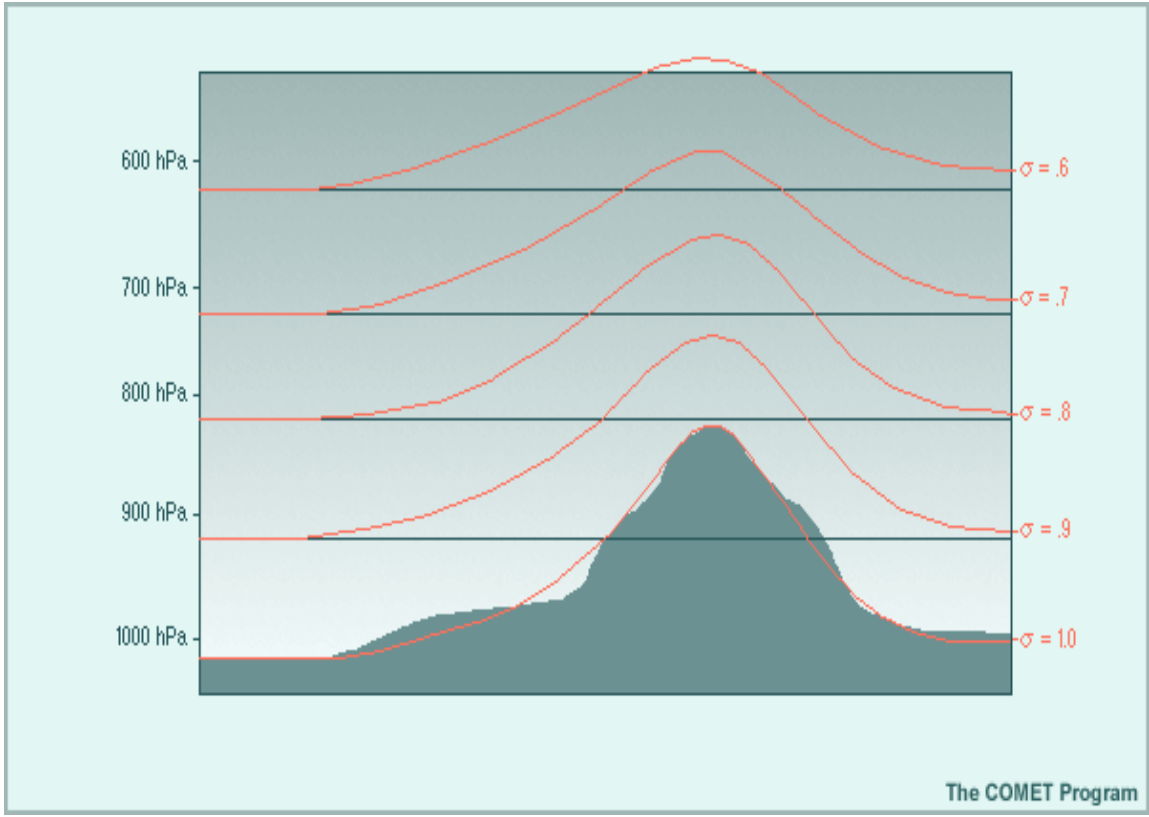
Al igual que la configuración horizontal, la estructura vertical es fundamental para establecer el comportamiento del modelo. Para poder representar apropiadamente la estructura vertical de la atmósfera es necesario seleccionar un sistema de coordenadas adecuado y una resolución vertical suficiente.

La expresión más simple de las ecuaciones de movimiento, las cuales constituyen la base de todos los modelos de predicción numérica, se obtiene en las coordenadas de presión.



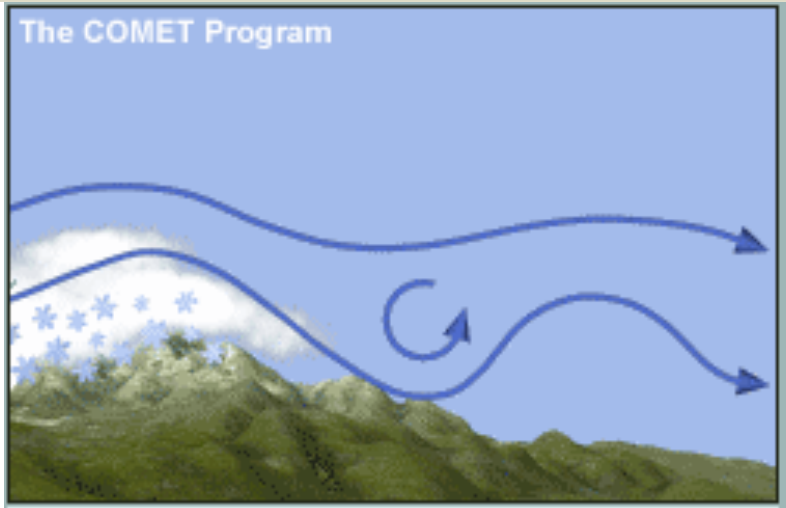
Coordenada vertical sigma

En 1957 se desarrolló la coordenada sigma (σ), que sigue el contorno del terreno. Aunque tradicionalmente la coordenada sigma se ha usado en la mayoría de los modelos climáticos globales, actualmente muchos modelos han adoptado sistemas híbridos en los cuales se emplea una coordenada sigma que sigue el contorno del terreno en los niveles bajos y se transforma en algún otro sistema de coordenadas en altura.

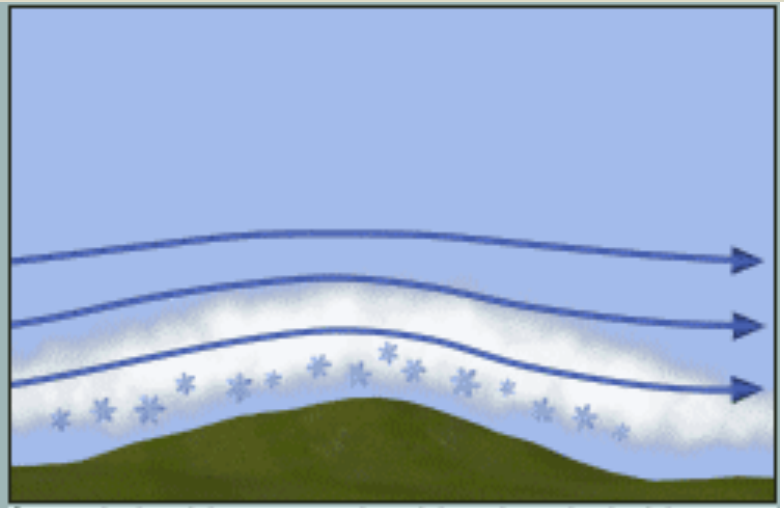


Topografía del modelo y topografía real

La representación de la topografía de la superficie por el modelo determina su capacidad de pronosticar las estructuras meteorológicas inducidas por el terreno.
Si un modelo no representa un accidente topográfico, no puede resolver o pronosticar un fenómeno meteorológico inducido o influenciado por esa estructura topográfica.



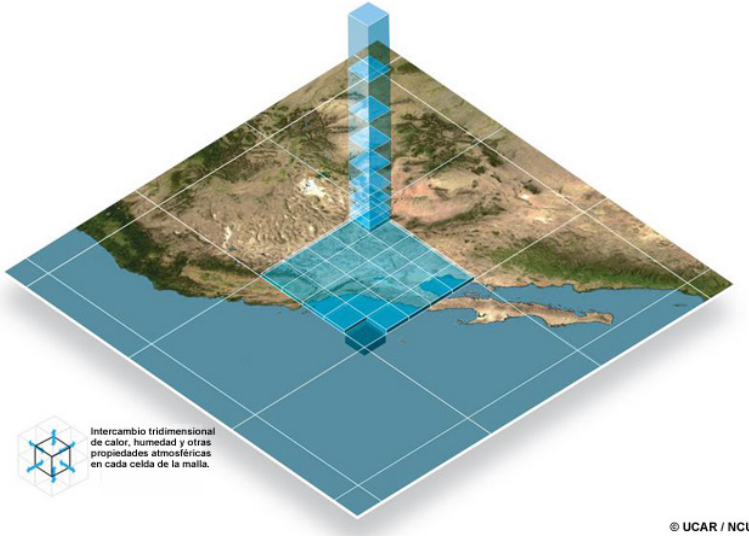
Superficie real con vientos pendiente abajo inducidos por el terreno, nubes de montaña, y precipitación.



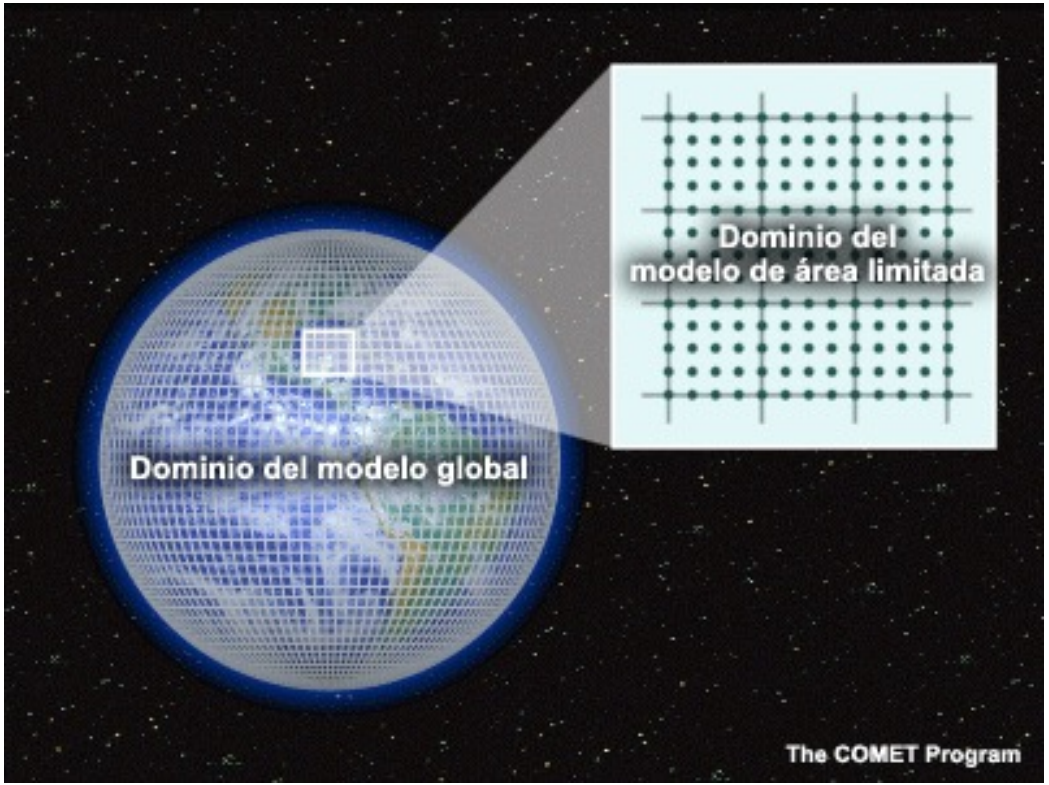
La resolución del terreno en el modelo y el suavizado del terreno en el pronóstico del modelo no le permiten para describir procesos físicos asociados con terrenos complejos o inclinados

MODELOS CLIMÁTICOS REGIONALES

Son modelos numéricos de area limitada, como por ejemplo, el area de Sudamérica o el Perú, llegando a resoluciones espaciales altas entre 25 – 1 km.
Estos modelos son alimentados por los datos de los modelos globales, para la condiciones iniciales y de contorno.

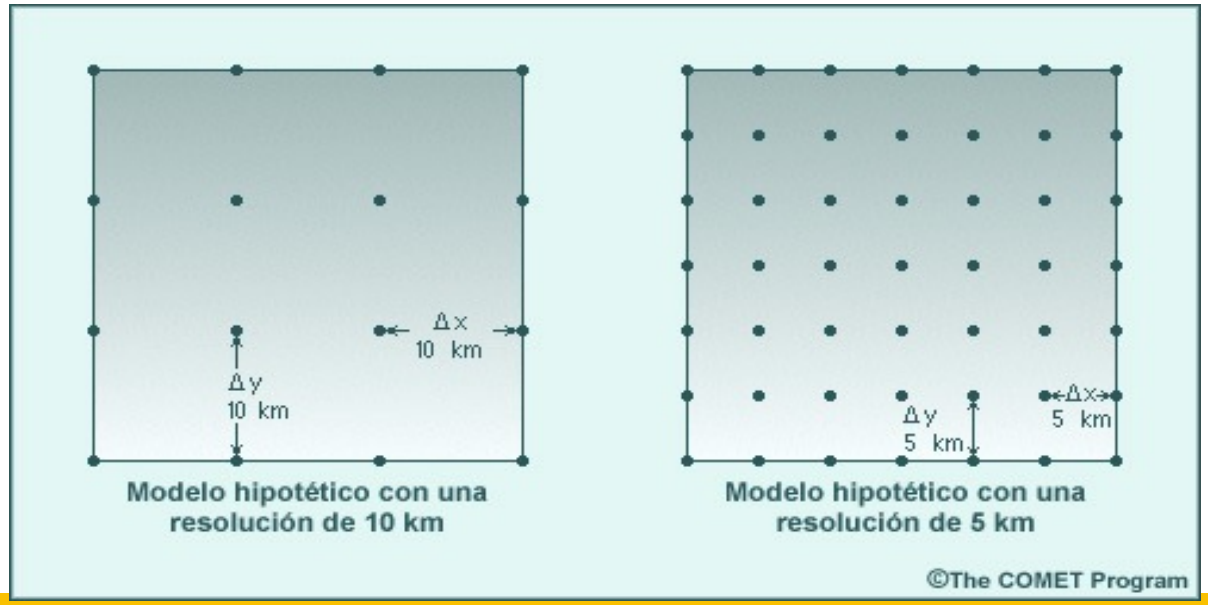


© UCAR / NCU

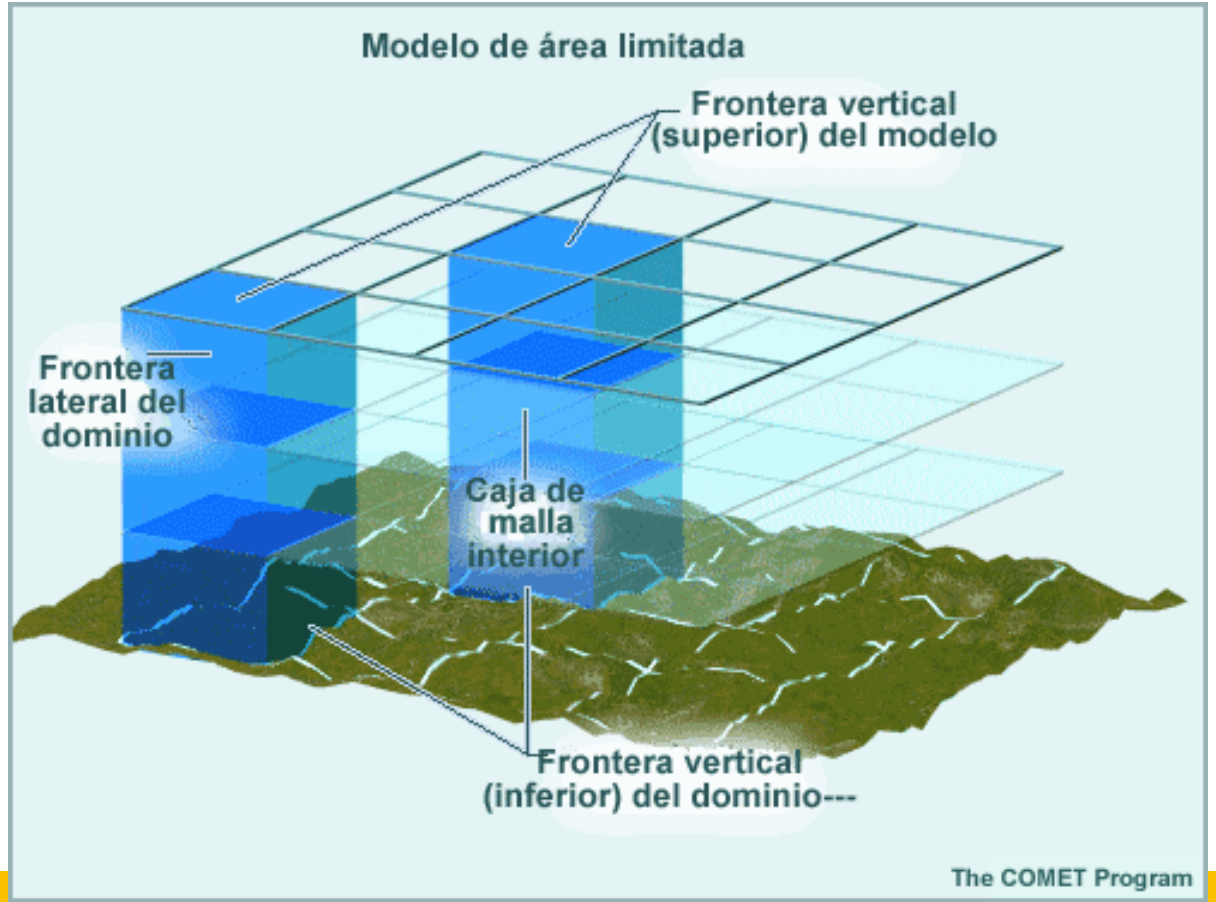


Aumento de resolución

Cuando se aumenta la resolución de un modelo, se incrementa la demanda en los recursos computacionales, porque el modelo debe determinar los valores para una mayor cantidad de puntos de malla. Como se ilustra a continuación, si reducimos la distancia entre los puntos de malla del modelo se reduce a la mitad (mayor resolución), el número de puntos de malla sobre la misma área se cuadruplica.



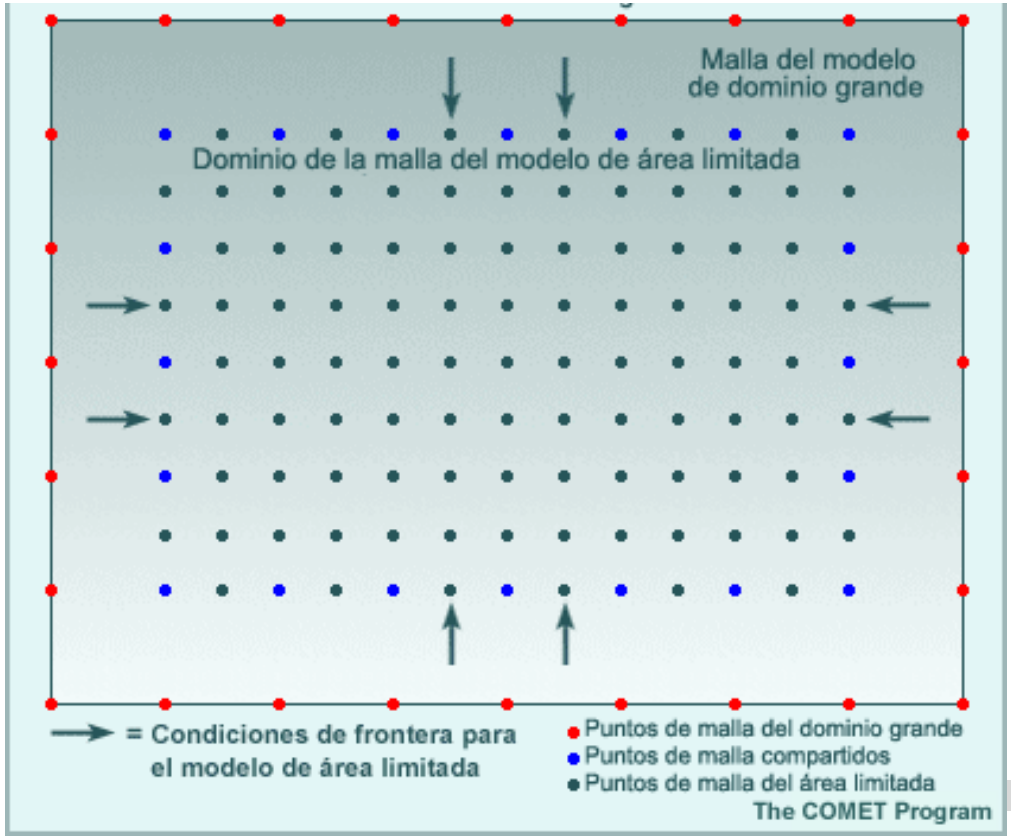
Dominio y condiciones de frontera



Dominio y condiciones de frontera

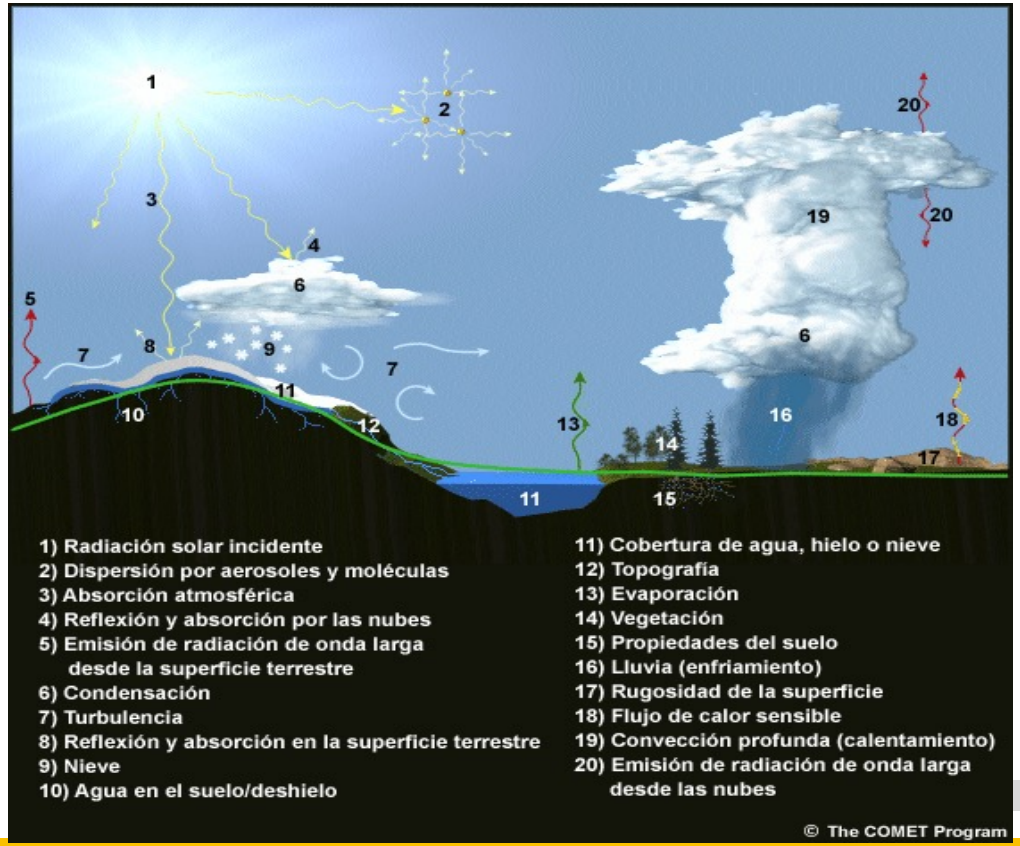
Los modelos de dominio mayor aportan los datos para las condiciones de fronteras laterales de los modelos de área limitada. Todos los modelos deben contar con información exacta de todas las variables de pronóstico en el dominio y a lo largo de cada frontera del modelo (lateral, superior e inferior) para poder resolver las ecuaciones de pronóstico. Los valores de frontera pueden provenir de varias fuentes, incluyendo:

- Asimilación de datos.
- valor climatológico o fijo, entre otras fuentes.



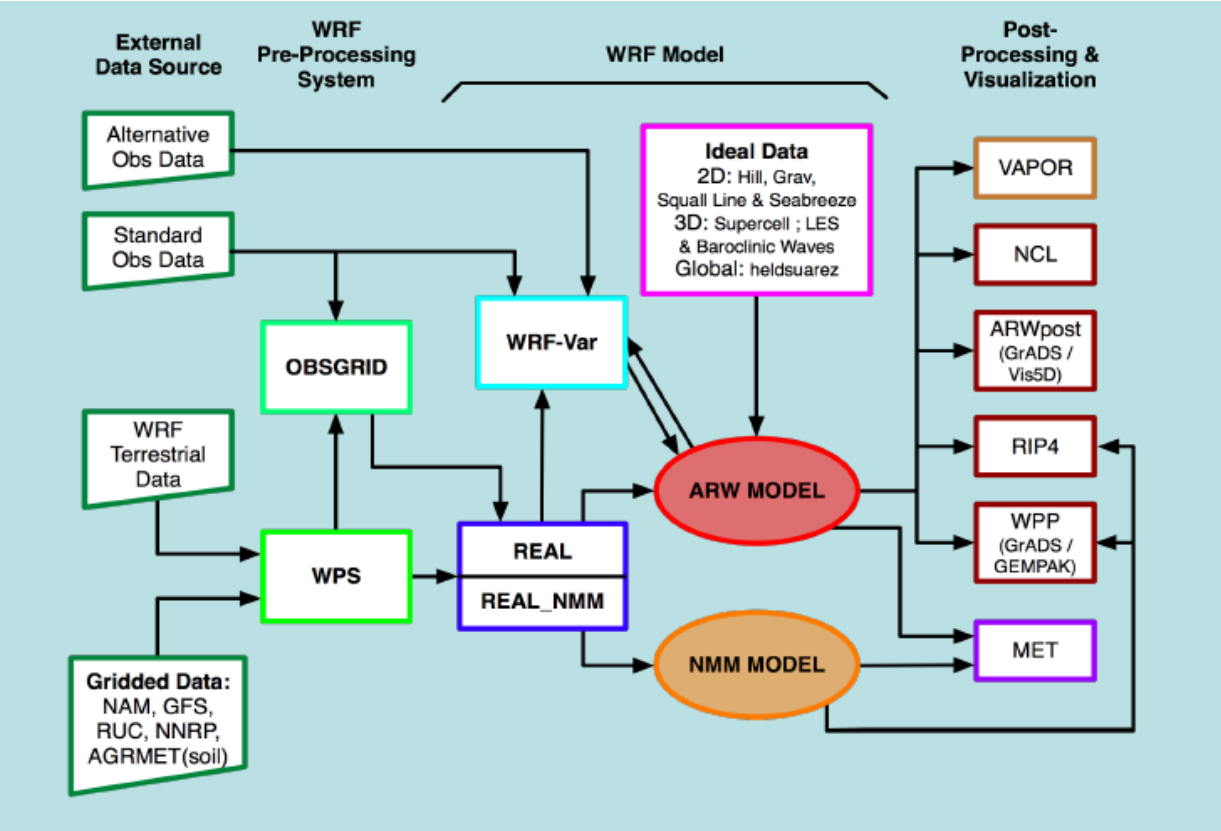
Parametrización

Esta figura representa algunos de los procesos físicos que se suelen parametrizar. Dado que estos procesos no se pueden pronosticar explícitamente y con todo detalle con las ecuaciones de pronóstico del modelo, sea cual sea su resolución, es necesario parametrizarlos.

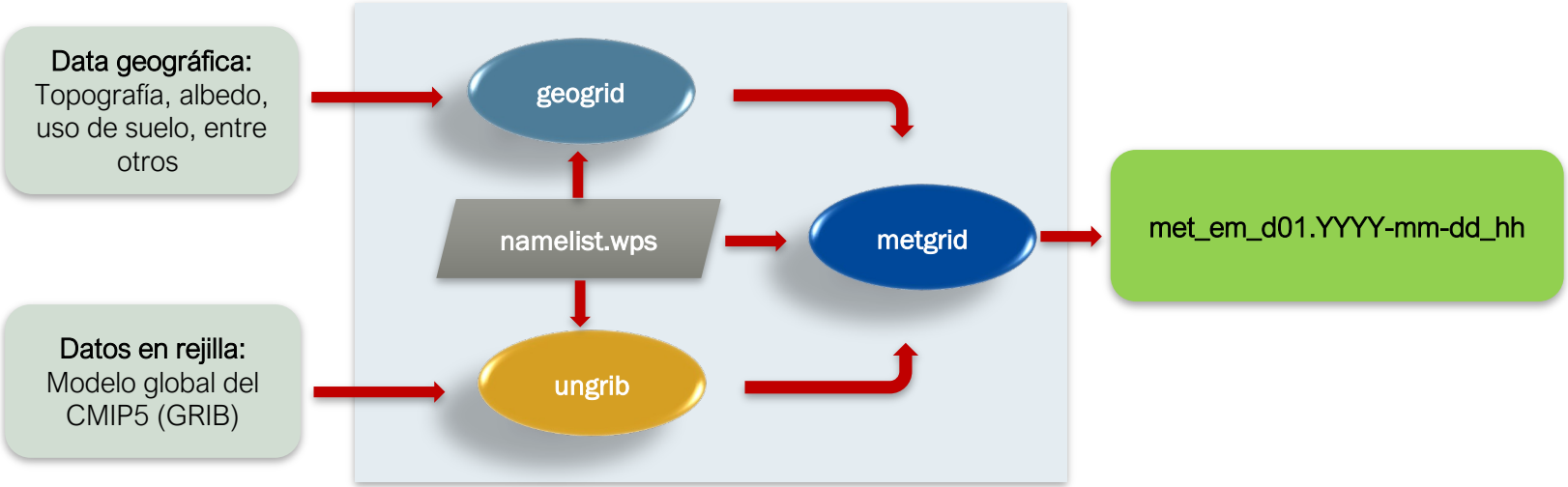


Modelo regional WRF

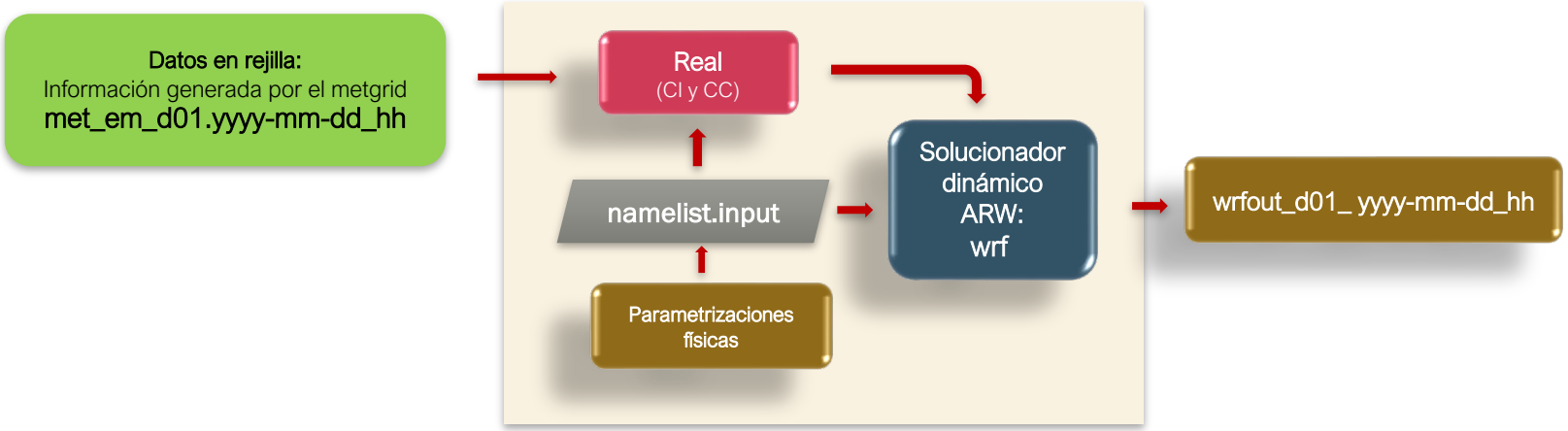
The Weather Research and Forecasting (WRF), es un modelo de área limitada de código abierto de nueva generación, diseñado tanto para cubrir necesidades de pronósticos operativos e investigación atmosférica.



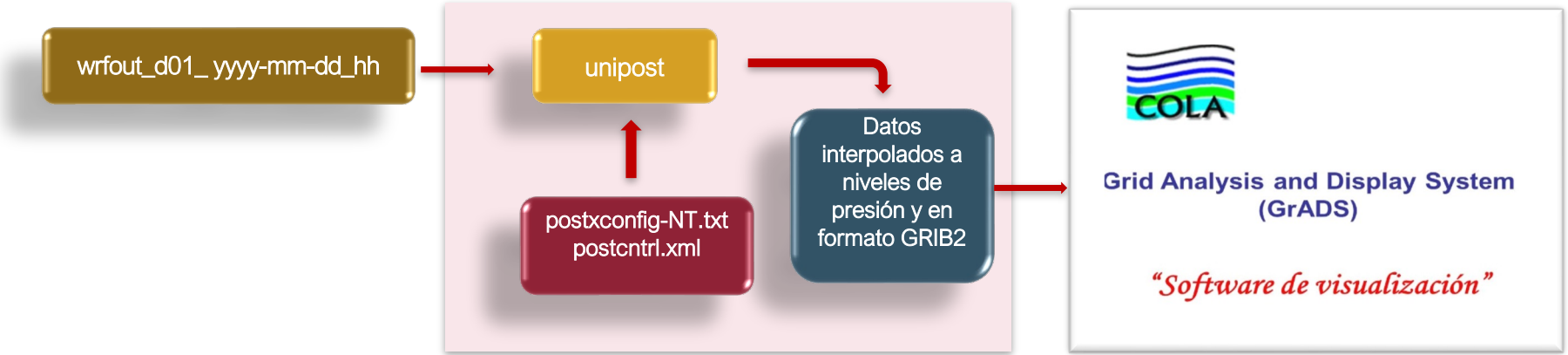
Modelo regional WRF: WPS



Modelo regional WRF: WRF



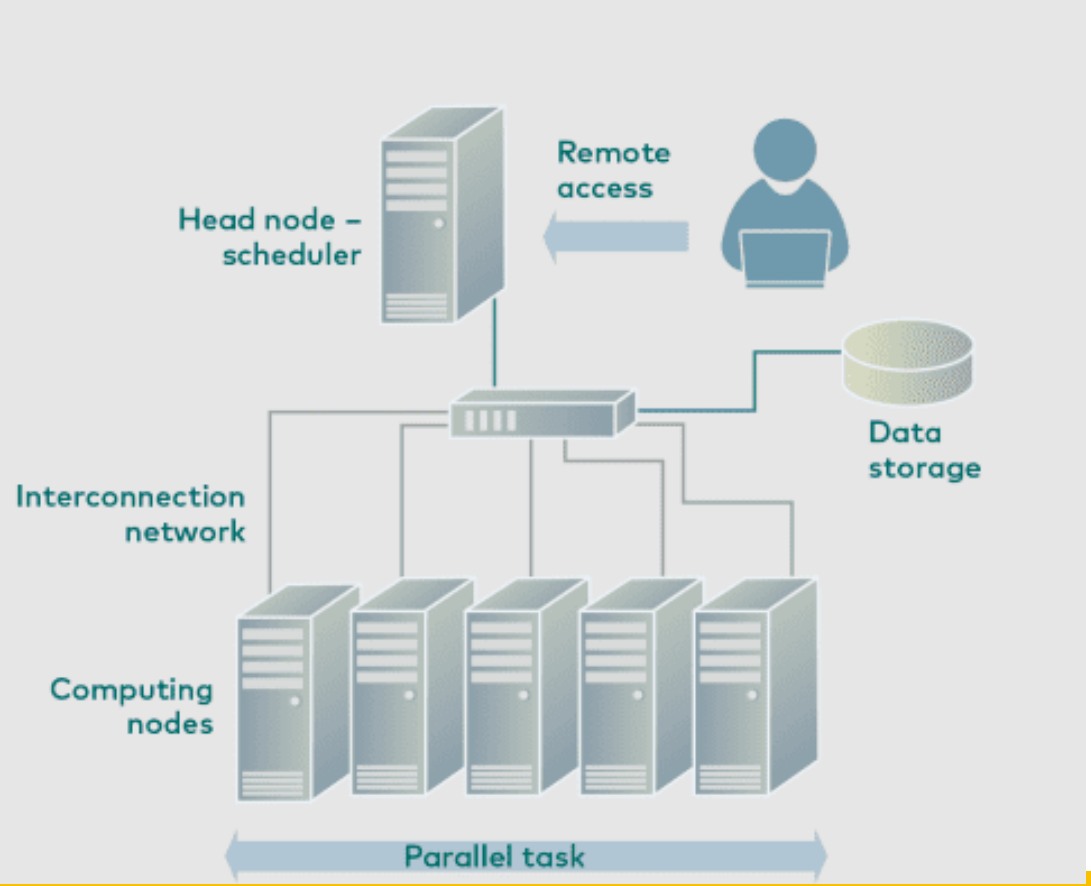
Modelo regional WRF: UPP



RECURSO COMPUTACIONAL

Computadora personal (CPU)	Cluster HPC
1. Numero de Cores 4 a 6 por procesador 2. Un (01) procesador físico por CPU 3. Total de Cores x CPU = 6 Cores 4. Demasiado tiempo para simular un domino de alta resolución espacial 5. Por ejemplo un dominio para la región Lima a 5 km de resolución por día de simulación demoraría 5 horas.	1. Numero de Cores hasta 24 por procesador 2. Hasta 4 procesadores físicos por servidor 3. Total de Cores x servidor = 96 Cores 4. Rapidez en el tiempo de simulación de un domino de alta resolución espacial 5. Para simular el mismo dominio para la región Lima a 5 km de resolución por día de simulación demoraría 10 minutos.

RECURSO COMPUTACIONAL: CLUSTER HPC



PERÚ Ministerio del Ambiente

Senamhi
SERVICIO NACIONAL DE METEOROLOGÍA E HIDROLOGÍA DEL PERÚ

Siempre con el pueblo

INSTALACIÓN DE LIBRERÍAS COMPILACIÓN DEL WRF

REQUERIMIENTO DEL SISTEMA

- ***¿SOBRE QUE TIPO DE SISTEMAS CORRE EL WRF?***

ARQUITECTURAS DE HARDWARE DE 32 Ó 64 BIT

SISTEMA OPERATIVO LINUX, MacOSX

- ***EJEMPLOS DE SISTEMAS ACEPTABLES:***

LAPTOPS, DESKTOPS Y CLUSTER CON LINUX

LAPTOPS, DESKTOPS Y CLUSTER CON MacOSX

REQUERIMIENTO DEL SISTEMA

Es indispensable tener un compilador en fortran y C instalados en el sistema operativo como:

- GNU -> gfortran, gcc, cpp
- Intel -> ifort, icc, icpc
- Pgi -> pgf90, pgcc, pgCC

Verificar la existencia de los compiladores

- which ifort
- which icc
- which icpc

Verificar la versión del compilador

- ifort --version

- export FC=ifort
- export F90=ifort
- export F77=ifort
- export CC=icc
- export CXX=icpc

LIBRERÍAS

- **NetCDF**: El formato NetCDF (Network Common Data Format) fue creado por UNIDATA como formato estándar para que sea usado en algunos de sus softwares que ofrece a la comunidad científica.
- **ZLIB**: es un software utilizado para compresión de datos.
- **JASPER**: se utiliza para compresión de imágenes.
- **LIBPNG**: libpng es la librería de compresión sin perdida
- **MPICH**: es un programa que se utiliza para aplicaciones de memoria distribuida que utilizan computación paralela

MUY IMPORTANTE!

Para instalar el WRF and WPS es necesario que sean instaladas con el mismo compilador que se usaron para las librerías MPICH2, NetCDF, JasPer, zlib, y libpng;

COMPILACIÓN DEL WRF

- tar –xzvf WRFV-4.3.TAR.gz
- cd WRF-4.3
- ./configure
- Seleccionar la opción 15

```
------------------------------------------------------------------------
Please select from among the following Linux x86_64 options:

  1. (serial)   2. (smpar)   3. (dmpar)   4. (dm+sm)   PGI (pgf90/gcc)
  5. (serial)   6. (smpar)   7. (dmpar)   8. (dm+sm)   PGI (pgf90/pgcc): SGI MPT
  9. (serial)  10. (smpar)  11. (dmpar)  12. (dm+sm)   PGI (pgf90/gcc): PGI accelerator
 13. (serial)  14. (smpar)  15. (dmpar)  16. (dm+sm)   INTEL (ifort/icc)
                                         17. (dm+sm)   INTEL (ifort/icc): Xeon Phi (MIC architecture)
 18. (serial)  19. (smpar)  20. (dmpar)  21. (dm+sm)   INTEL (ifort/icc): Xeon (SNB with AVX mods)
 22. (serial)  23. (smpar)  24. (dmpar)  25. (dm+sm)   INTEL (ifort/icc): SGI MPT
 26. (serial)  27. (smpar)  28. (dmpar)  29. (dm+sm)   INTEL (ifort/icc): IBM POE
 30. (serial)               31. (dmpar)                PATHSCALE (pathf90/pathcc)
 32. (serial)  33. (smpar)  34. (dmpar)  35. (dm+sm)   GNU (gfortran/gcc)
 36. (serial)  37. (smpar)  38. (dmpar)  39. (dm+sm)   IBM (xlf90_r/cc_r)
 40. (serial)  41. (smpar)  42. (dmpar)  43. (dm+sm)   PGI (ftn/gcc): Cray XC CLE
 44. (serial)  45. (smpar)  46. (dmpar)  47. (dm+sm)   CRAY CCE (ftn/cc): Cray XE and XC
 48. (serial)  49. (smpar)  50. (dmpar)  51. (dm+sm)   INTEL (ftn/icc): Cray XC
 52. (serial)  53. (smpar)  54. (dmpar)  55. (dm+sm)   PGI (pgf90/pgcc)
 56. (serial)  57. (smpar)  58. (dmpar)  59. (dm+sm)   PGI (pgf90/gcc): -f90=pgf90
 60. (serial)  61. (smpar)  62. (dmpar)  63. (dm+sm)   PGI (pgf90/pgcc): -f90=pgf90
 64. (serial)  65. (smpar)  66. (dmpar)  67. (dm+sm)   INTEL (ifort/icc): HSW/BDW
 68. (serial)  69. (smpar)  70. (dmpar)  71. (dm+sm)   INTEL (ifort/icc): KNL MIC

Enter selection [1-71] : 15
------------------------------------------------------------------------
Compile for nesting? (1=basic, 2=preset moves, 3=vortex following) [default 1]: 1
```

COMPILACIÓN DEL WRF

- configure.wrf

```
DESCRIPTION     =       INTEL ($SFC/$SCC)
DMPARALLEL      =        1
OMPCPP          =       # -D_OPENMP
OMP             =       # -openmp -fpp -auto
OMPCC           =       # -openmp -fpp -auto
SFC             =       ifort
SCC             =       icc
CCOMP           =       icc
DM_FC           =       mpif90 -f90=$(SFC)
DM_CC           =       mpicc -cc=$(SCC)
FC              =       time $(DM_FC)
CC              =       $(DM_CC)
LD              =       $(FC)
RWORDSIZE       =       $(NATIVE_RWORDSIZE)
PROMOTION       =       -real-size `expr 8 \* $(RWORDSIZE)` -i4
ARCH LOCAL      =       -DNONSTANDARD SYSTEM FUNC  -DWRF USE CLM
```

COMPILACIÓN DEL WRF

- Compilar el modelo WRF
- ./compile em_real >& compil.log & -> (3d real case)

Compilación satisfactoria se generan los principales ejecutables

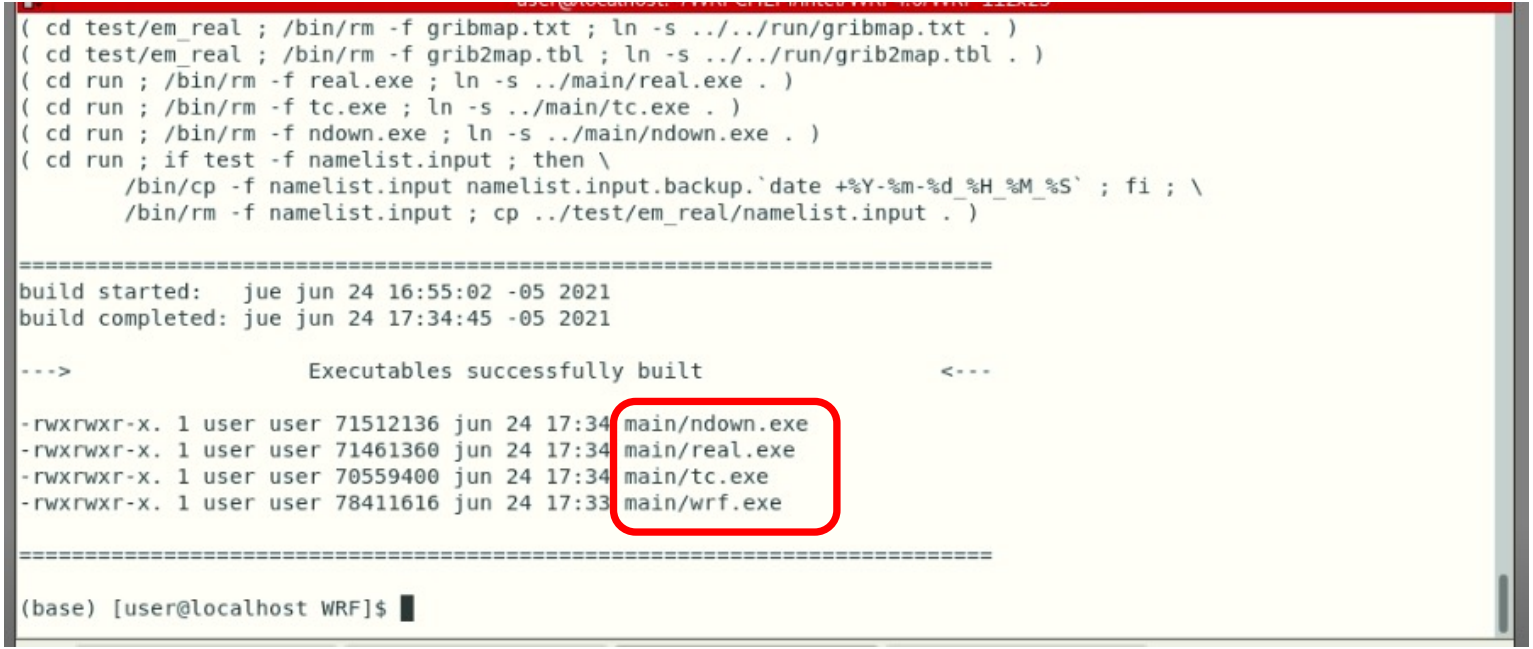
wrf.exe -> ejecutable del modelo

real.exe -> Inicialización de los datos CI y CC

estos ejecutables son linkeados desde WRF-4.3/main a dos carpetas WRF-4.3/run y WRF-4.3/test/em_real

COMPILACIÓN DEL WRF

- compile em_real



```
( cd test/em_real ; /bin/rm -f gribmap.txt ; ln -s ../../run/gribmap.txt . )
( cd test/em_real ; /bin/rm -f grib2map.tbl ; ln -s ../../run/grib2map.tbl . )
( cd run ; /bin/rm -f real.exe ; ln -s ../main/real.exe . )
( cd run ; /bin/rm -f tc.exe ; ln -s ../main/tc.exe . )
( cd run ; /bin/rm -f ndown.exe ; ln -s ../main/ndown.exe . )
( cd run ; if test -f namelist.input ; then \
        /bin/cp -f namelist.input namelist.input.backup.`date +%Y-%m-%d_%H_%M_%S` ; fi ; \
        /bin/rm -f namelist.input ; cp ../test/em_real/namelist.input . )

=========================================================================
build started:   jue jun 24 16:55:02 -05 2021
build completed: jue jun 24 17:34:45 -05 2021

--->                  Executables successfully built                  <---

-rwxrwxr-x. 1 user user 71512136 jun 24 17:34 main/ndown.exe
-rwxrwxr-x. 1 user user 71461360 jun 24 17:34 main/real.exe
-rwxrwxr-x. 1 user user 70559400 jun 24 17:34 main/tc.exe
-rwxrwxr-x. 1 user user 78411616 jun 24 17:33 main/wrf.exe

=========================================================================

(base) [user@localhost WRF]$
```

COMPILACIÓN DEL WPS

```
(base) [user@localhost WPS]$ ls
arch    compile    geogrid       metgrid       namelist.wps.all_options  namelist.wps.global  README  util
clean   configure  link_grib.csh  namelist.wps  namelist.wps.fire         namelist.wps.nmm     ungrib
(base) [user@localhost WPS]$ ./configure


  13.  Linux x86_64, IA64 and Opteron    (serial)
  14.  Linux x86_64, IA64 and Opteron    (serial_NO_GRIB2)
  15.  Linux x86_64, IA64 and Opteron    (dmpar)
  16.  Linux x86_64, IA64 and Opteron    (dmpar_NO_GRIB2)
  17.  Linux x86_64, Intel compiler    (serial)
  18.  Linux x86_64, Intel compiler    (serial_NO_GRIB2)
  19.  Linux x86_64, Intel compiler    (dmpar)
  20.  Linux x86_64, Intel compiler    (dmpar_NO_GRIB2)
  21.  Linux x86_64, Intel compiler, SGI MPT    (serial)
  22.  Linux x86_64, Intel compiler, SGI MPT    (serial_NO_GRIB2)
  23.  Linux x86_64, Intel compiler, SGI MPT    (dmpar)
  24.  Linux x86_64, Intel compiler, SGI MPT    (dmpar_NO_GRIB2)
  25.  Linux x86_64, Intel compiler, IBM POE    (serial)
  26.  Linux x86_64, Intel compiler, IBM POE    (serial_NO_GRIB2)
  27.  Linux x86_64, Intel compiler, IBM POE    (dmpar)
  28.  Linux x86_64, Intel compiler, IBM POE    (dmpar_NO_GRIB2)
  29.  Linux x86_64 g95 compiler     (serial)
  30.  Linux x86_64 g95 compiler     (serial_NO_GRIB2)
  31.  Linux x86_64 g95 compiler     (dmpar)
  32.  Linux x86_64 g95 compiler     (dmpar_NO_GRIB2)
  33.  Cray XE/XC CLE/Linux x86_64, Cray compiler   (serial)
  34.  Cray XE/XC CLE/Linux x86_64, Cray compiler   (serial_NO_GRIB2)
  35.  Cray XE/XC CLE/Linux x86_64, Cray compiler   (dmpar)
  36.  Cray XE/XC CLE/Linux x86_64, Cray compiler   (dmpar_NO_GRIB2)
  37.  Cray XC CLE/Linux x86_64, Intel compiler   (serial)
  38.  Cray XC CLE/Linux x86_64, Intel compiler   (serial_NO_GRIB2)
  39.  Cray XC CLE/Linux x86_64, Intel compiler   (dmpar)
  40.  Cray XC CLE/Linux x86_64, Intel compiler   (dmpar_NO_GRIB2)

Enter selection [1-40] : 19
```

Se escoge la opción 19 compilador Intel

COMPILACIÓN DEL WPS

```
(base) [user@localhost WPS]$ ls
arch     configure      link_grib.csh  namelist.wps.all_options  namelist.wps.nmm  util
clean    configure.wps  metgrid        namelist.wps.fire         README
compile  geogrid        namelist.wps   namelist.wps.global       ungrib
(base) [user@localhost WPS]$ ./compile



        CPPFLAGS="-D_UNDERSCORE -DBYTESWAP -DLINUX -DIO_NETCDF -DIO_BINARY -DIO_GRIB1 -DBIT32 -D_MPI -D_UTIL" )
make[1]: se ingresa al directorio `/home/user/WRFCHEM/intel/WRF4.0/WPS/util/src'
/bin/rm -f int2nc.f90
/lib/cpp -P -traditional -D_UNDERSCORE -DBYTESWAP -DLINUX -DIO_NETCDF -DIO_BINARY -DIO_GRIB1 -DBIT32 -D_MPI -D_U
TIL int2nc.F > int2nc.f90
ifort -FR -convert big_endian -I/home/user/WRFCHEM/intel/WRF4.0/WPS/../WRF/external/io_netcdf -I/home/user/WRFCH
EM/intel/WRF4.0/WPS/../WRF/external/io_grib_share -I/home/user/WRFCHEM/intel/WRF4.0/WPS/../WRF/external/io_grib1
 -I/home/user/WRFCHEM/intel/WRF4.0/WPS/../WRF/external/io_int -I/home/user/WRFCHEM/intel/WRF4.0/WPS/../WRF/inc -
I/home/user/WRFCHEM/intel/installWRFCHEM/intelsoft/include -c int2nc.f90
ifort -o int2nc.exe  -I/home/user/WRFCHEM/intel/WRF4.0/WPS/../WRF/external/io_netcdf -I/home/user/WRFCHEM/intel/
WRF4.0/WPS/../WRF/external/io_grib_share -I/home/user/WRFCHEM/intel/WRF4.0/WPS/../WRF/external/io_grib1 -I/home/
user/WRFCHEM/intel/WRF4.0/WPS/../WRF/external/io_int -I/home/user/WRFCHEM/intel/WRF4.0/WPS/../WRF/inc -I/home/us
er/WRFCHEM/intel/installWRFCHEM/intelsoft/include int2nc.o module_debug.o misc_definitions_module.o read_met_mod
ule.o cio.o -L/home/user/WRFCHEM/intel/WRF4.0/WPS/../WRF/external/io_grib1 -lio_grib1 -L/home/user/WRFCHEM/intel
/WRF4.0/WPS/../WRF/external/io_grib_share -lio_grib_share -L/home/user/WRFCHEM/intel/WRF4.0/WPS/../WRF/external/
io_int -lwrfio_int -L/home/user/WRFCHEM/intel/WRF4.0/WPS/../WRF/external/io_netcdf -lwrfio_nf -L/home/user/WRFCH
EM/intel/installWRFCHEM/intelsoft/lib  -lnetcdf
make[1]: se sale del directorio `/home/user/WRFCHEM/intel/WRF4.0/WPS/util/src'
if [ -h int2nc.exe ] ; then \
        /bin/rm -f int2nc.exe ; \
fi ; \
if [ -h ../int2nc.exe ] ; then \
        /bin/rm -f ../int2nc.exe ; \
fi ; \
if [ -e src/int2nc.exe ] ; then \
        ln -sf src/int2nc.exe . ; \
fi
(base) [user@localhost WPS]$ ls
arch     configure      geogrid.exe    metgrid.exe              namelist.wps.fire    README      util
clean    configure.wps  link_grib.csh  namelist.wps             namelist.wps.global  ungrib
compile  geogrid        metgrid        namelist.wps.all_options  namelist.wps.nmm     ungrib.exe
(base) [user@localhost WPS]$
```

Si la compilación se realizó satisfactoriamente se tendría que visualizar los siguientes ejecutables: geogrid.exe, ungrib.exe y metgrid.exe

COMPILACIÓN DEL UPP

```
configure: making ./bin
bindir  /home/user/WRFCHEM/intel/WRF4.0/UPPV3.2/bin
configure: making ./include
incmod  /home/user/WRFCHEM/intel/WRF4.0/UPPV3.2/include
configure: making ./lib
libdir  /home/user/WRFCHEM/intel/WRF4.0/UPPV3.2/lib
JASPER Environent found :: GRIB2 library ::
grib2lib = -L/home/user/WRFCHEM/intel/installWRFCHEM/intelsoft/lib -lpng -lz -ljasper
grib2inc = -I/home/user/WRFCHEM/intel/installWRFCHEM/intelsoft/include
------------------------------------------------------------------------
Please select from among the following supported platforms.

   1.  Linux x86_64, PGI compiler  (serial)
   2.  Linux x86_64, PGI compiler  (dmpar)
   3.  Linux x86_64, Intel compiler      (serial)
   4.  Linux x86_64, Intel compiler      (dmpar)
   5.  Linux x86_64, Intel compiler, SGI MPT     (serial)
   6.  Linux x86_64, Intel compiler, SGI MPT     (dmpar)
   7.  Linux x86_64, gfortran compiler  (serial)
   8.  Linux x86_64, gfortran compiler  (dmpar)
   9.  Linux x86_64, Intel compiler, IBM POE     (serial)
  10.  Linux x86_64, Intel compiler, IBM POE     (dmpar)
  11.  Linux x86_64, gfortran compiler: -f90=gfortran  (serial)
  12.  Linux x86_64, gfortran compiler: -f90=gfortran  (dmpar)
  13.  Linux x86_64, PGI compiler: -f90=pgf90  (serial)
  14.  Linux x86_64, PGI compiler: -f90=pgf90  (dmpar)

Enter selection [1-14] : 4
------------------------------------------------------------------------
Configuration successful. To build the UPP, type: compile
------------------------------------------------------------------------
(base) [user@localhost UPPV3.2]$
```

```
(base) [user@localhost UPPV3.2]$ ls
arch  bin  clean  compile  configure  configure.upp  include  lib  makefile  parm  scripts  src
(base) [user@localhost UPPV3.2]$ ./compile
```

Luego que el módulo UPPV3.2 se ha instalado satisfactoriamente se generan los siguientes ejecutables: copygb.exe, ndate.exe y unipost.exe

PERÚ Ministerio del Ambiente

Senamhi
SERVICIO NACIONAL DE METEOROLOGÍA E HIDROLOGÍA DEL PERÚ

Siempre con el pueblo

Dominio de simulación
Domain Wizard

Dominio de simulación

U.S. Department of Commerce | National Oceanic & Atmospheric Administration | NOAA Research

Earth System Research Laboratory
Global Systems Division

Search ESRL: Search
Calendar | People | Publications

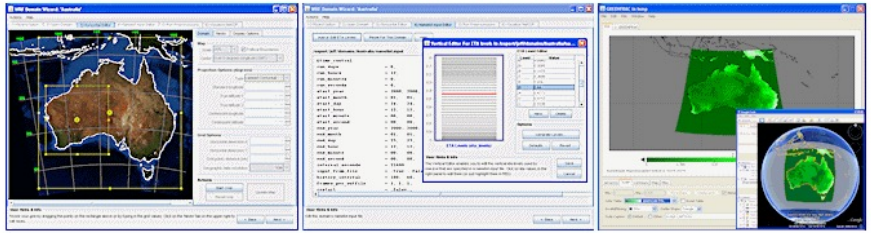
WRF Portal

- Home
- WRF Portal
- Domain Wizard
- FIM Portal
- Tutorials (HTML)
- Tutorials (Video)
- F.A.Q.
- About the WRF Team at GSD
- Contacts at GSD

WRF Domain Wizard

GUI for the WRF Preprocessor System (WPS) and namelist.input. Go here for LAPS Version

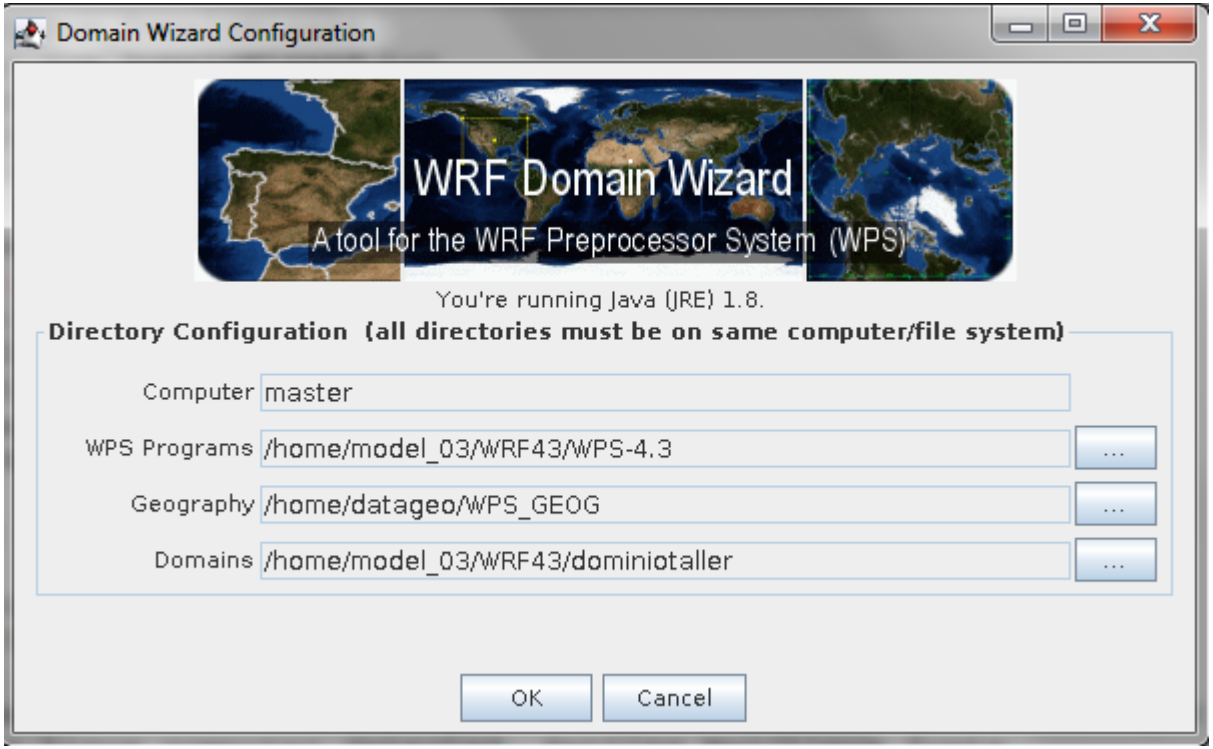
Version 2.84 for Linux, AIX, Mac, and Windows - released August 15, 2013



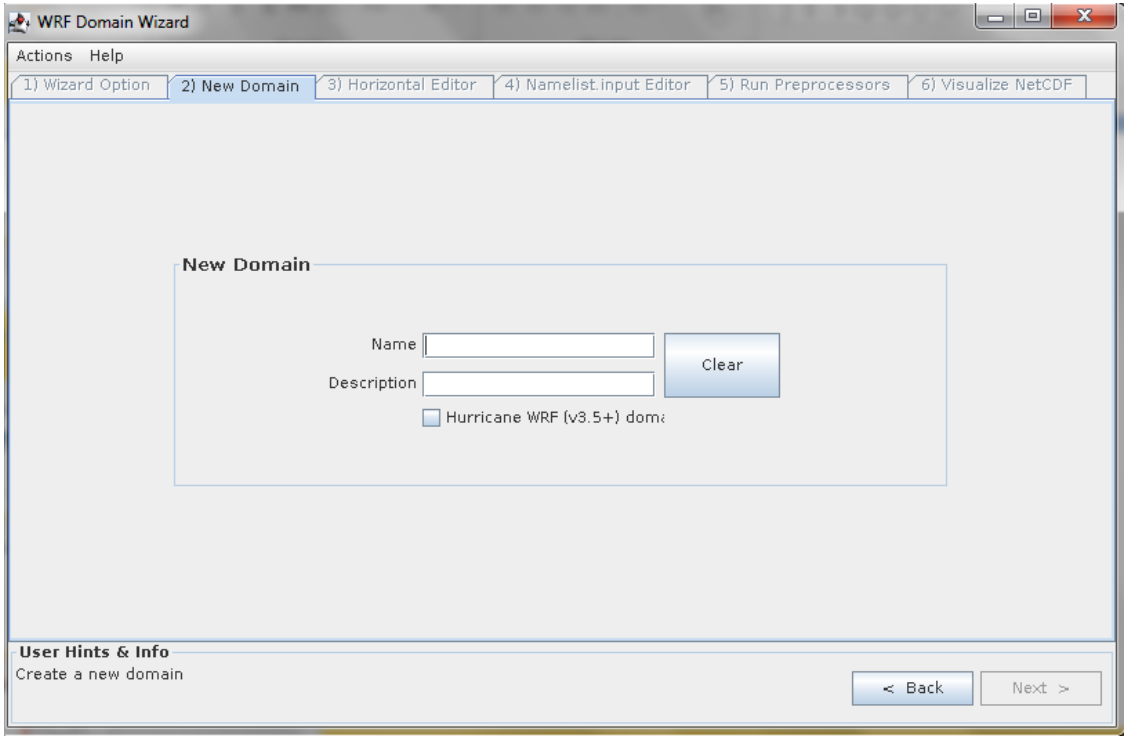
WRF Domain Wizard is the successor to the WRFSI GUI and is a graphical user interface (GUI) for the new WRF Preprocessing System (WPS). It enables users to easily define and localize domains (cases) by selecting a region of the Earth and choosing a map projection. Users can also define nests using the nests editor, edit namelist.input, run the WPS programs (geogrid, ungrib, and metgrid) through the GUI, and visualize the NetCDF output. WRF Domain Wizard is also a built-in component of WRF Portal. WRF Domain Wizard stores its information in namelist.wps, namelist.input (and nest7grid.parms for LAPS users).

https://esrl.noaa.gov/gsd/wrfportal/DomainWizard.html

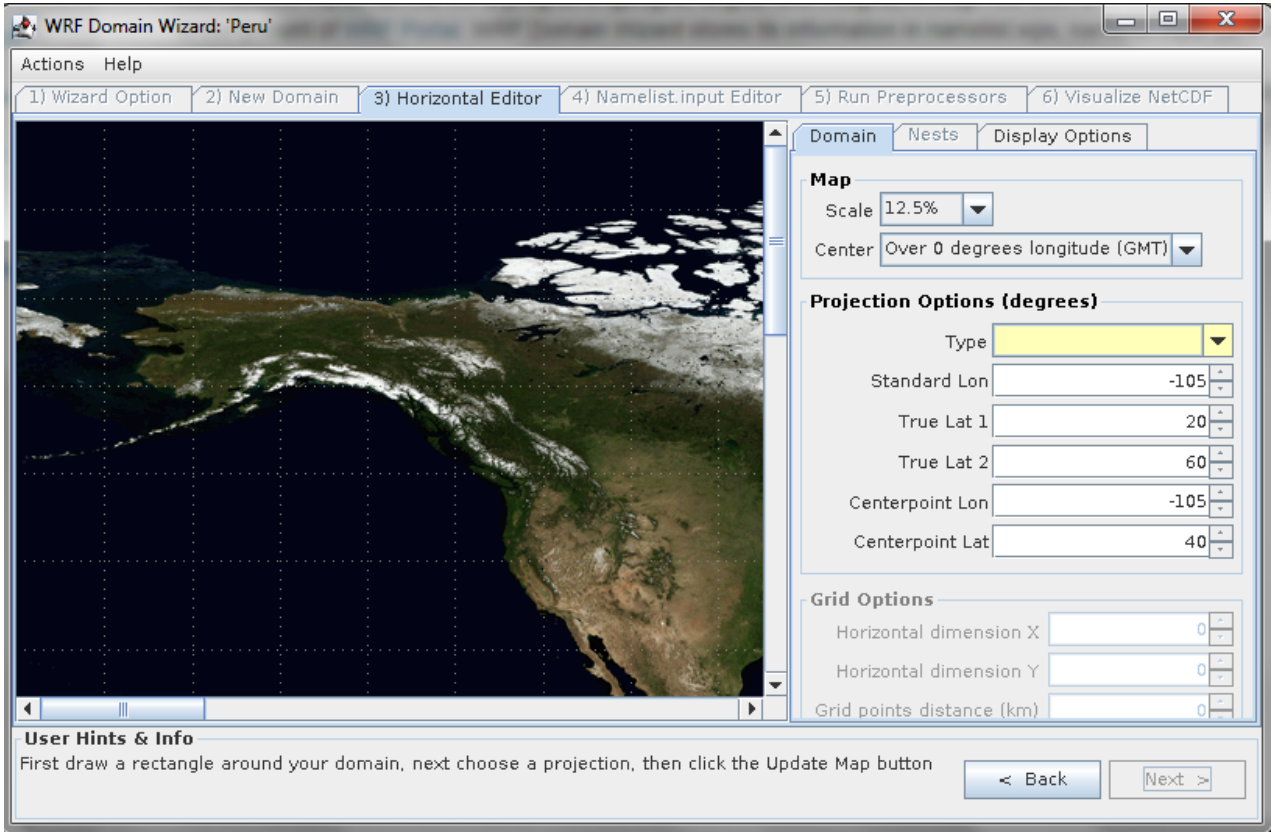
Dominio de simulación



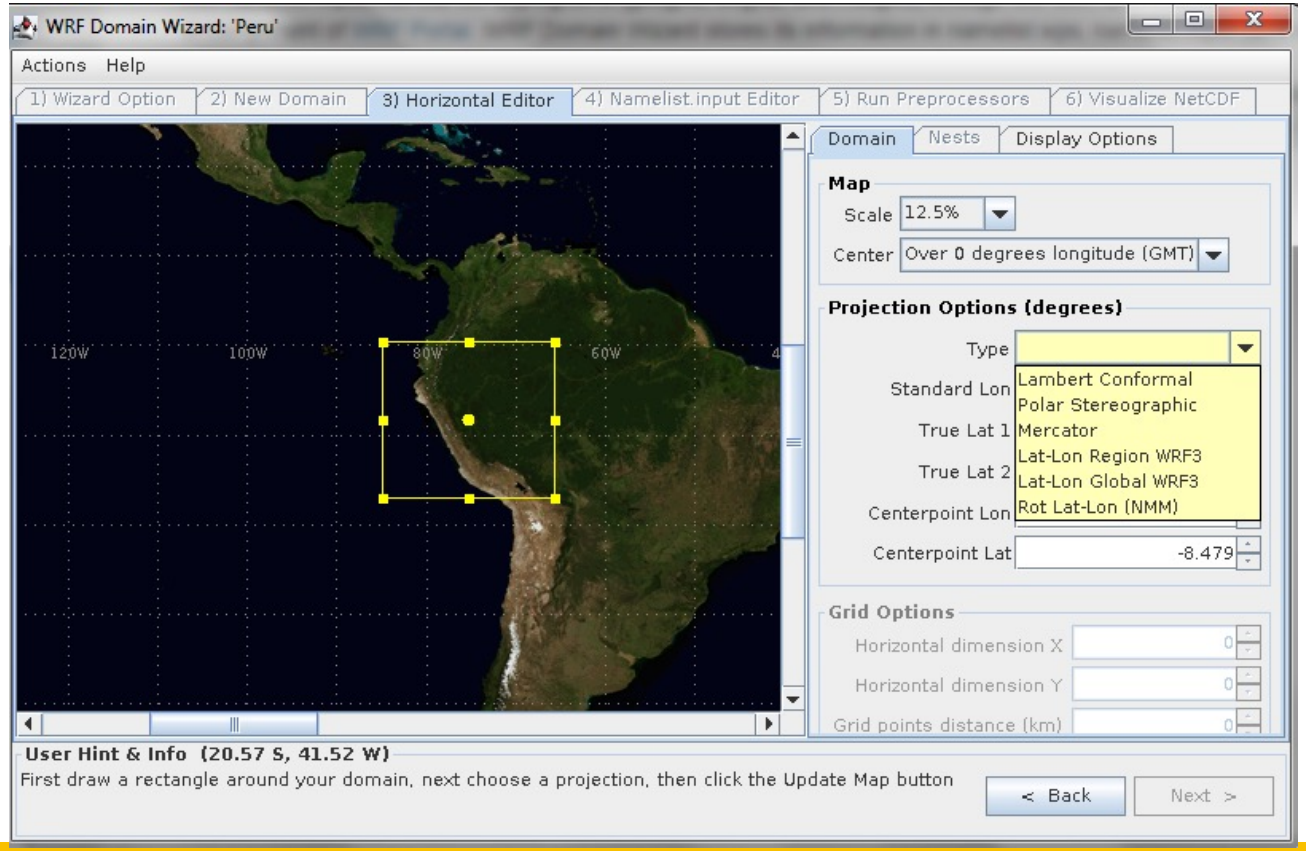
Dominio de simulación



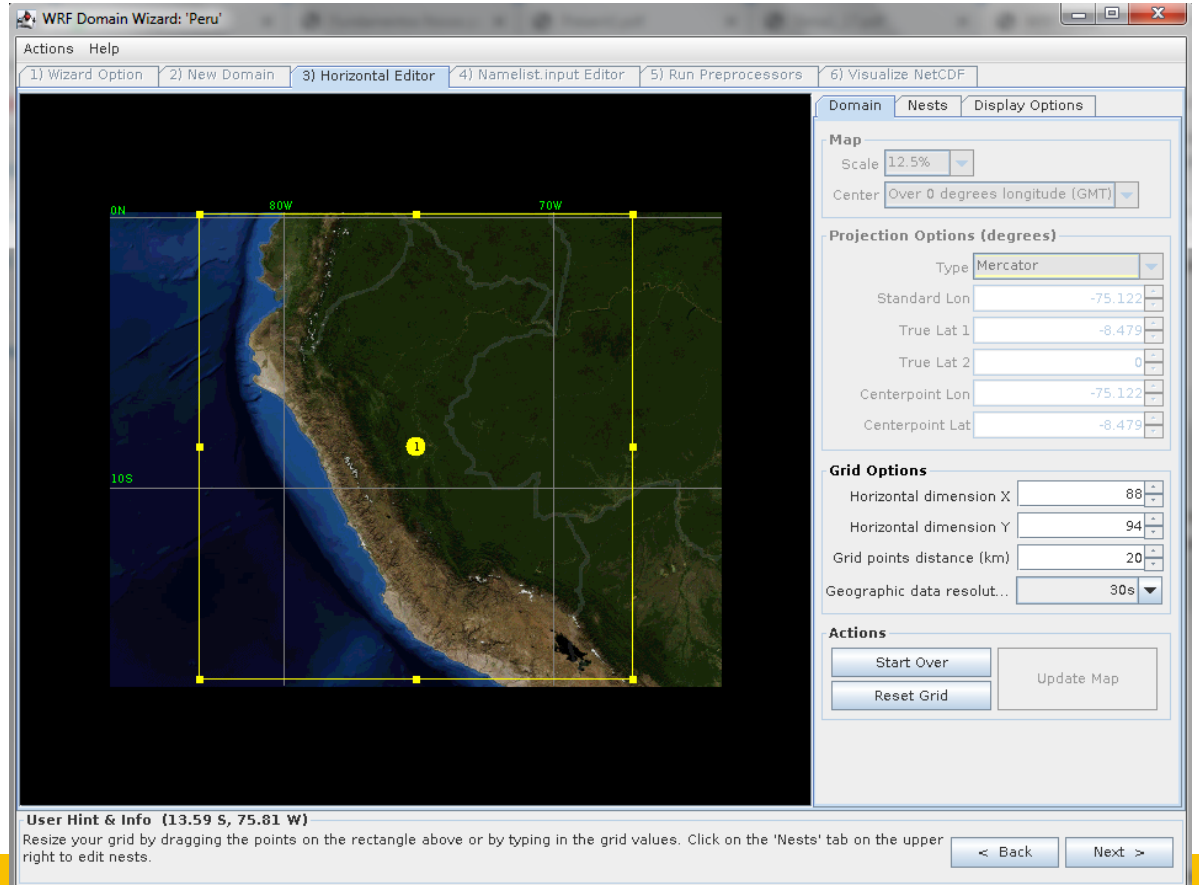
Dominio de simulación



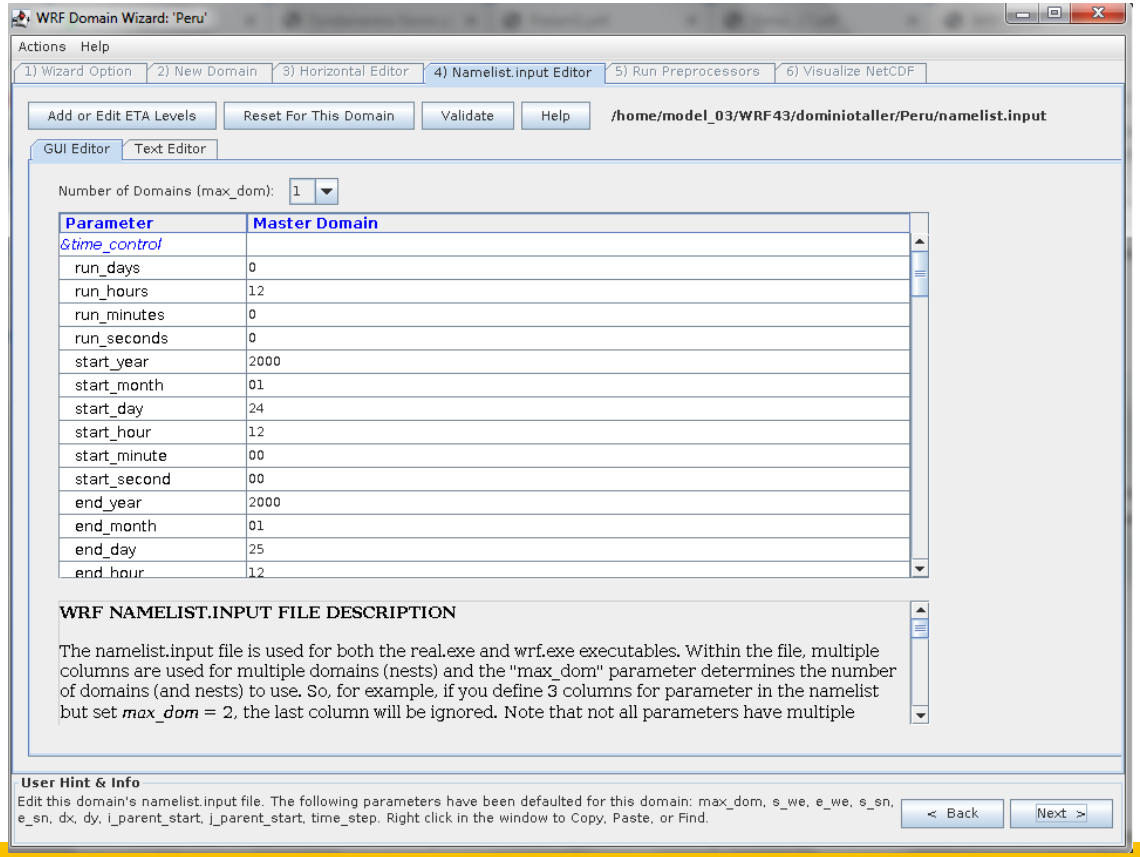
Dominio de simulación



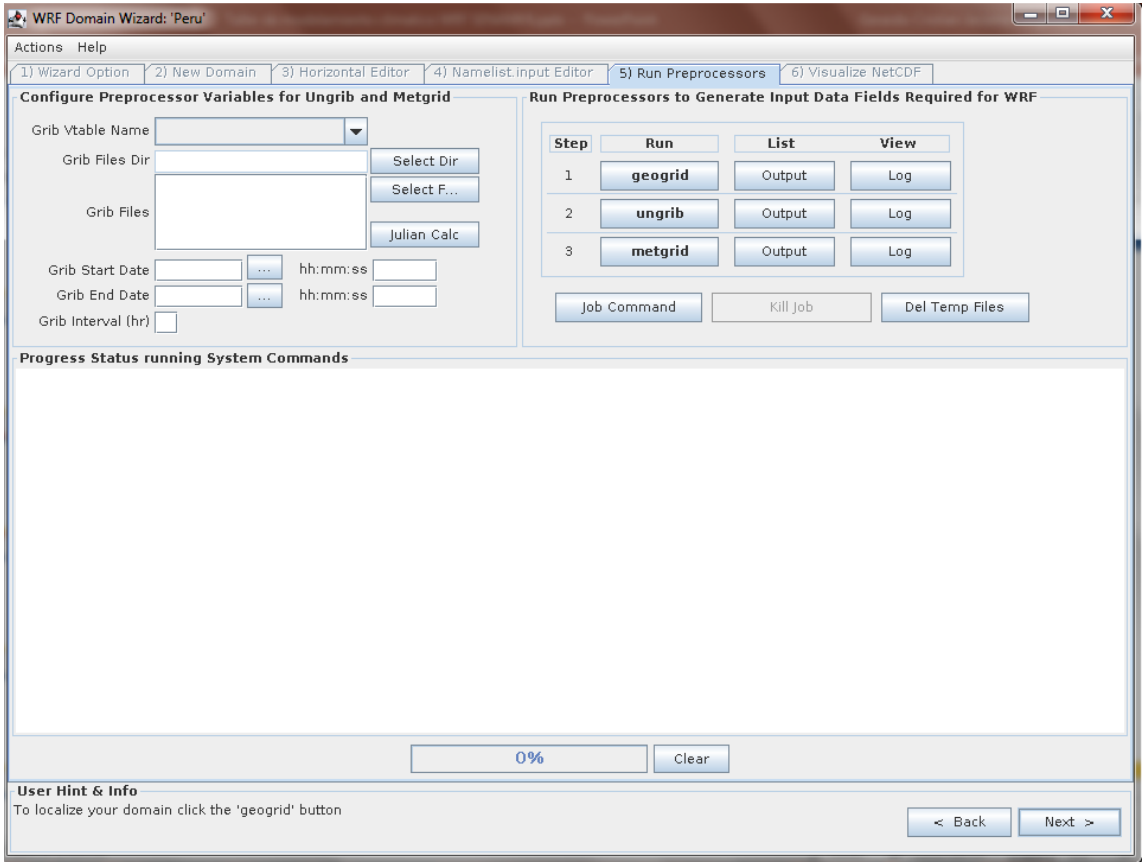
Dominio de simulación



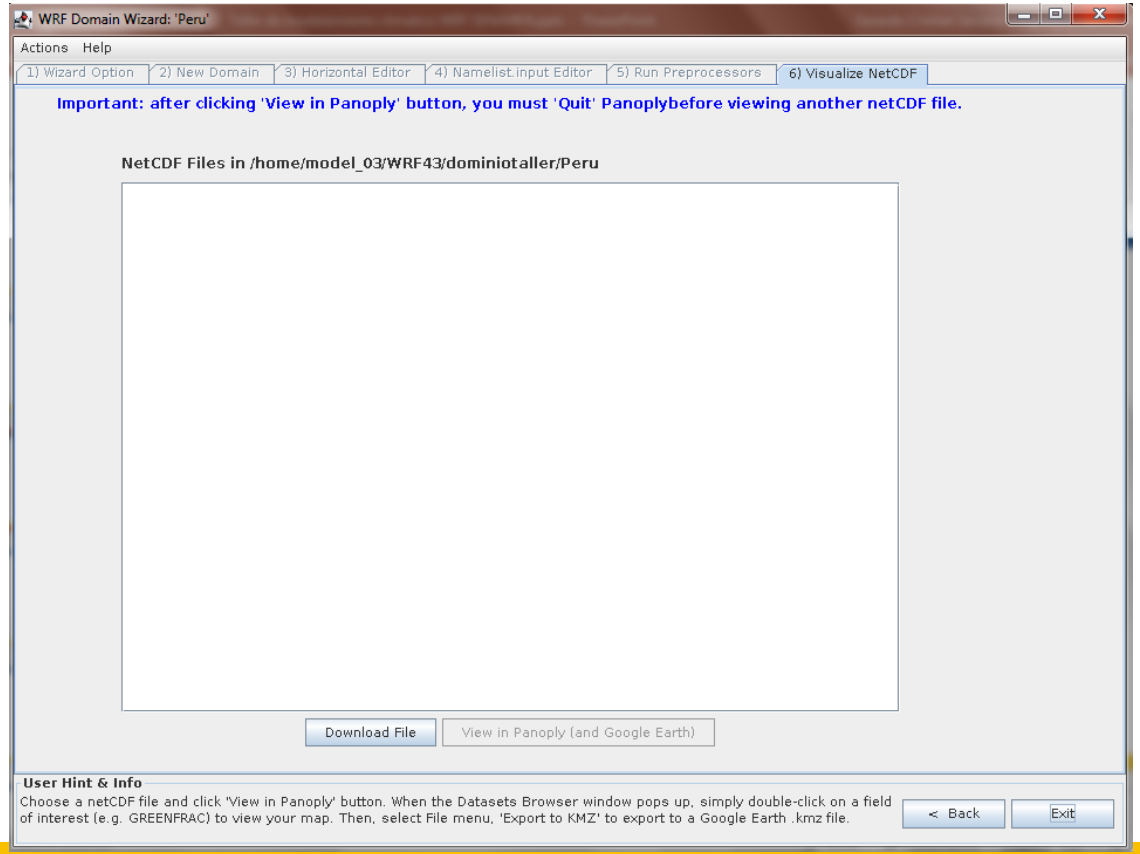
Dominio de simulación



Dominio de simulación



Dominio de simulación

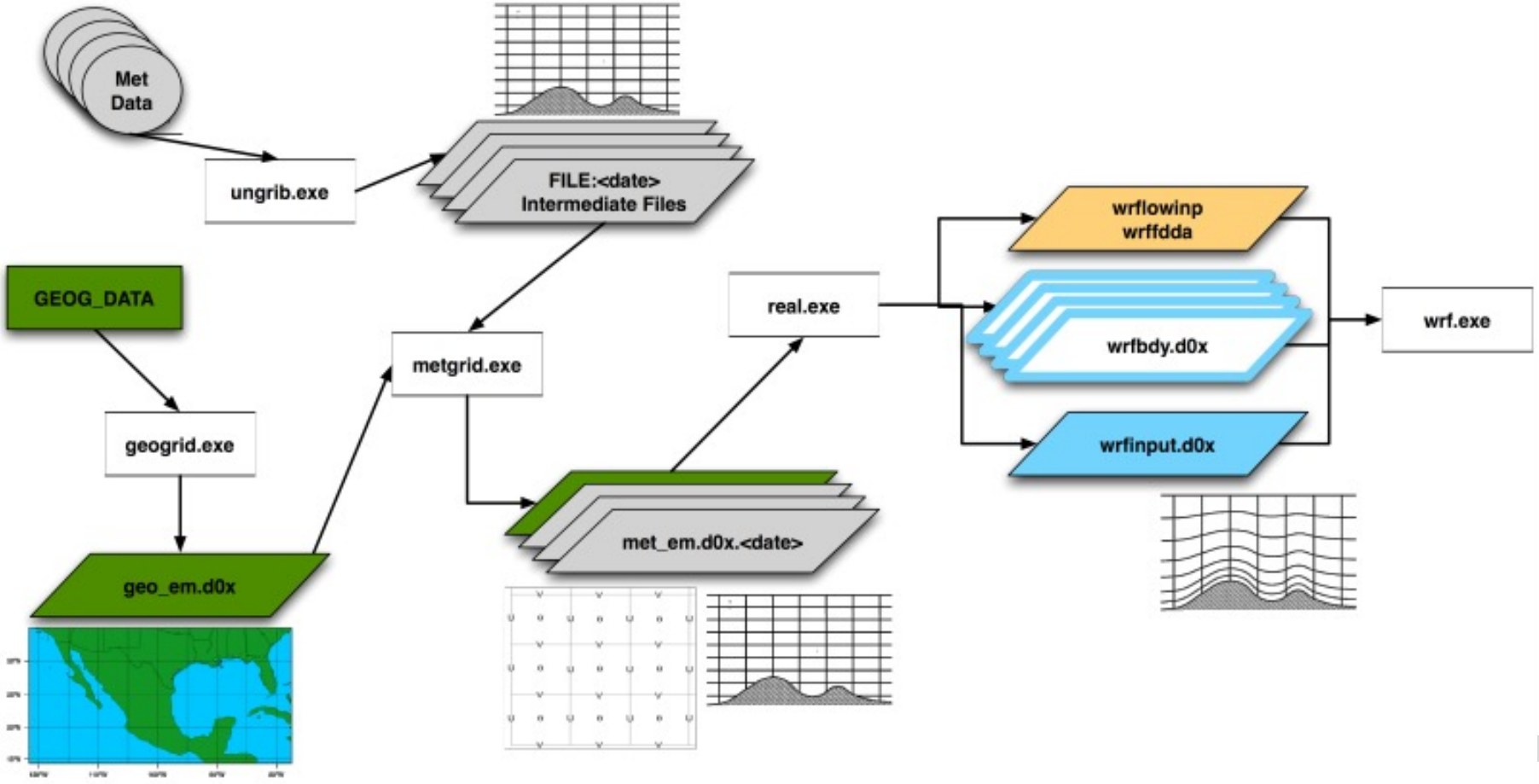


PERÚ Ministerio del Ambiente

Senamhi
SERVICIO NACIONAL DE METEOROLOGÍA E HIDROLOGÍA DEL PERÚ

Siempre con el pueblo

EJECUCIÓN DEL MODELO WRF


Met Data
ungrib.exe
FILE:<date> Intermediate Files
GEOG_DATA
geogrid.exe
geo_em.d0x
metgrid.exe
met_em.d0x.<date>
real.exe
wrflowinp wrffdda
wrfbdy.d0x
wrfinput.d0x
wrf.exe

Ejecución del modelo WPS: namelist.wps

```
&share
 wrf_core = 'ARW',
 max_dom = 2,
 start_date = '2022-06-10_00:00:00', '2022-06-10_00:00:00',
 end_date   = '2022-06-15_00:00:00', '2022-06-15_00:00:00',
 interval_seconds = 21600,
 io_form_geogrid = 2,
 opt_output_from_geogrid_path = '/home/model_03/WRF43/WPS-4.3/',
 debug_level = 0,
/
```

Inicio de simulación

Fin de la simulación

Intervalo de los datos de entrada

Ejecución del modelo WPS: namelist.wps

```
&geogrid
 parent_id         = 1,1,
 parent_grid_ratio = 1,5,
 i_parent_start    = 1,22,
 j_parent_start    = 1,19,
 e_we          = 100,196,
 e_sn          = 68,151,
 geog_data_res = '30s','30s',
 dx = 50000,
 dy = 50000,
 map_proj =  'mercator',
 ref_lat   = -9.546,
 ref_lon   = -66.767,
 geog_data_path = '/home/datageo/WPS_GEOG/',
 opt_geogrid_tbl_path = '/home/model_03/WRF43/WPS-4.3/geogrid/',
/
```

→ Número de rejillas en x

→ Número de rejillas en y

→ Resolución espacial

Ejecución del modelo WPS: namelist.wps

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

&metgrid
 fg_name = 'FILE',
 io_form_metgrid = 2,
 opt_output_from_metgrid_path = '/home/model_03/WRF43/WPS-4.3/',
 opt_metgrid_tbl_path = '/home/model_03/WRF43/WPS-4.3/metgrid/',
/
```

Formato intermedio de los datos en GRIB (modelo global)

Ruta de los ficheros met_em_d01

Ejecución del modelo WPS: namelist.wps

Para ejecutar el preprocesamiento ingresar a la carpeta:

```
cd /donde/se/compilo/WRF4.3/WPS
```

Editar el fichero `namelist.wps` colocando la fecha de inicio de simulación y la fecha de finalización, seguidamente ejecutar el siguiente script:

```
./link_grib.csh /descarga/de/datos/GFS/*
```

Luego ejecutar el ungrib.exe, para lo cual en consola de Linux se ejecuta de la siguiente manera:

```
./ungrib.exe
```

El resultado son ficheros que tienen la siguiente sintaxis:

```
FILE:2016-10-01_00
FILE:2016-10-01_06
        .
        .
        .
FILE:2017-04-30_12
FILE:2017-04-30_18
```

Ejecución del modelo WPS: namelist.wps

Luego ejecutar metgrid.exe, para lo cual en consola de Linux se ejecuta de la siguiente manera:

```
./metgrid.exe
```

Se crearán los ficheros que tienen este formato `met_em_d01_YYYY-MM-DD-HH:00:00.nc`

```
met_em_d01_2016-10-01_00:00:00.nc
met_em_d01_2016-10-01_06:00:00.nc
          .
          .
          .
met_em_d01_2017-04-30_12:00:00.nc
met_em_d01_2017-04-30_18:00:00.nc
```

Ejecución del modelo WRF: namelist.input

```
&time_control
run_days                            = 0,
run_hours                           = 0,
run_minutes                         = 0,
run_seconds                         = 0,
start_year                          = 2016,     2022,
start_month                         = 10,       01,
start_day                           = 01,       02,
start_hour                          = 00,       00,
start_minute                        = 00,       00,
start_second                        = 00,       00,
end_year                            = 2017,     2022,
end_month                           = 05,       01,
end_day                             = 01,       02,
end_hour                            = 18,       12,
end_minute                          = 00,       00,
end_second                          = 00,       00,
interval_seconds                    = 21600,
input_from_file                     = .true.,   .true.,
history_interval                    = 180,       180,
frames_per_outfile                  = 1,        1,
restart                             = .false.,
restart_interval                    = 7200,
```

Inicio de simulación

Fin de la simulación

Intervalo de salida en minutos

Ejecución del modelo WRF: namelist.input

```
&domains
time_step                  = 10,
time_step_fract_num        = 0,
time_step_fract_den        = 1,
max_dom                    = 1,
e_we                       = 140,      136,
e_sn                       = 100,      116,
e_vert                     = 53,       53,
p_top_requested            = 5000,
num_metgrid_levels         = 38,
num_metgrid_soil_levels    = 4,
dx                         = 5000,     1000,
dy                         = 5000,     1000,
grid_id                    = 1,        2,
parent_id                  = 1,        1,
i_parent_start             = 1,       56,
j_parent_start             = 1,       36,
```

Paso de tiempo

Numero de dominios a simular

de rejillas a simular

Tamaño de la rejilla

Ejecución del modelo WRF: namelist.input

```
&physics
mp_physics                          = 10,        6,
ra_lw_physics                       = 4,         1,
ra_sw_physics                       = 4,         1,
radt                                = 16,        10,
sf_sfclay_physics                   = 91,        1,
sf_surface_physics                  = 2,         2,
bl_pbl_physics                      = 1,         1,
bldt                                = 0,         0,
cu_physics                          = 0,         0,
```

→ Microfísica de nubes

→ Radiación de onda larga

→ Radiación de onda corta

→ Capa limite planetaria

→ cúmulos

Ejecución del modelo WRF: namelist.input

- **REAL**

] ulimit –s unlimited

] ./real.exe

CI: wrfinput_d01

CC: wrfbdy_d01 y wrflowinp_d01

- **WRF**

] time mpirun –np 6 ./wrf.exe

wrfout_d01_2016-10-01_00:00:00

wrfout_d01_2016-10-01_06:00:00

.

wrfout_d01_2017-04-30_21:00:00

PERÚ Ministerio del Ambiente

Senamhi
SERVICIO NACIONAL DE METEOROLOGÍA E HIDROLOGÍA DEL PERÚ

Siempre con el pueblo

Acceso al servidor de SENAMHI

Acceso al servidor de SENAMHI

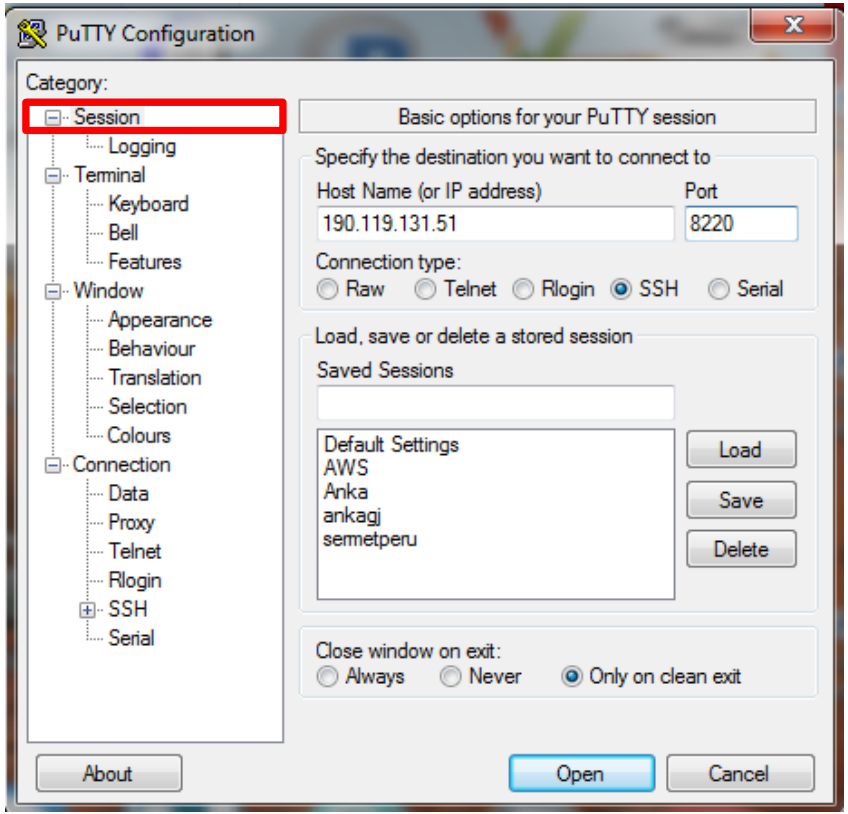
Credenciales de acceso

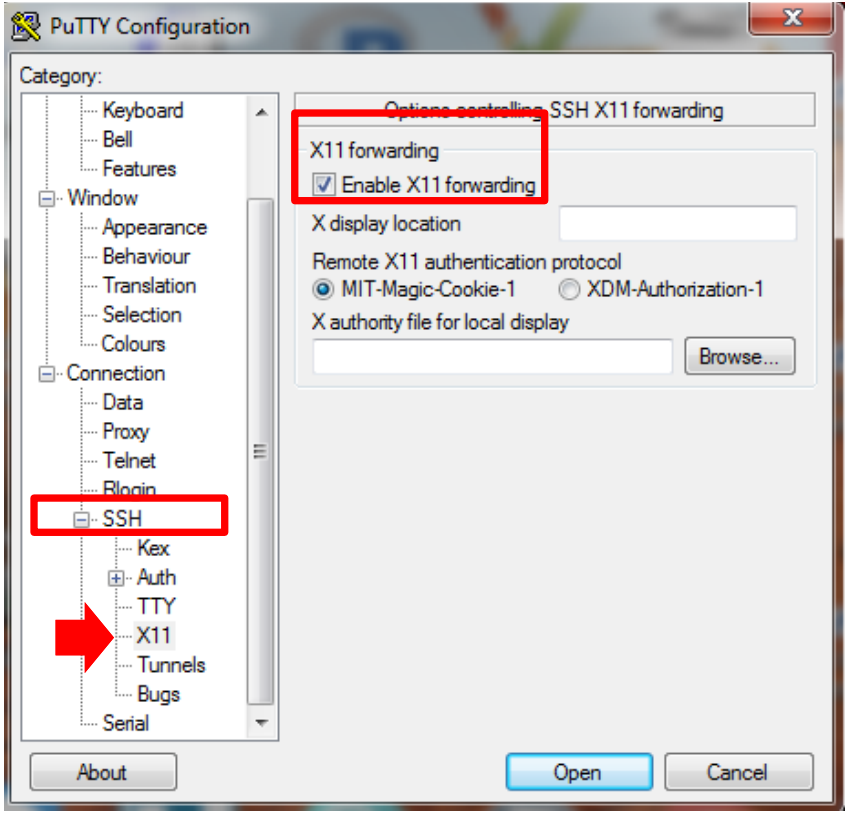
- IP: 190.119.131.51
- PUERTO: 8220
- USUARIO: prueba0, prueba1 y prueba2
- PASSW: capacitacion2k22

Shearmely, Florez Chaccara Joel Yoel Alania Sumaran Imelda Valentina Aliaga Guerreros	GRUPO1	prueba0
Carmen Rosa Farfan Tovar Janet Huaman Vargas Berly Berrios Dueñas	GRUPO2	prueba1
Gabriel Leví Caro-Sánchez Gago Irving Jesus Lizarraga Isla Hector Yauri Quispe Yureisse Marian Barrueta Faching	GRUPO3	prueba2

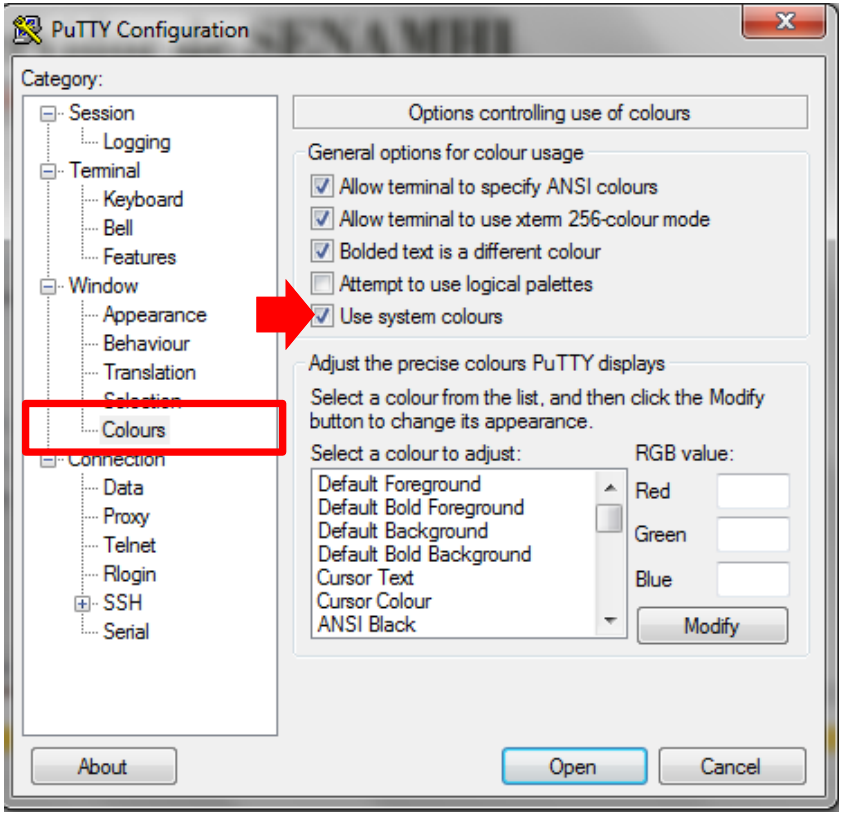
- Putty (https://www.chiark.greenend.org.uk/~sgtatham/putty/latest.html)
- Xming (https://sourceforge.net/projects/xming/)

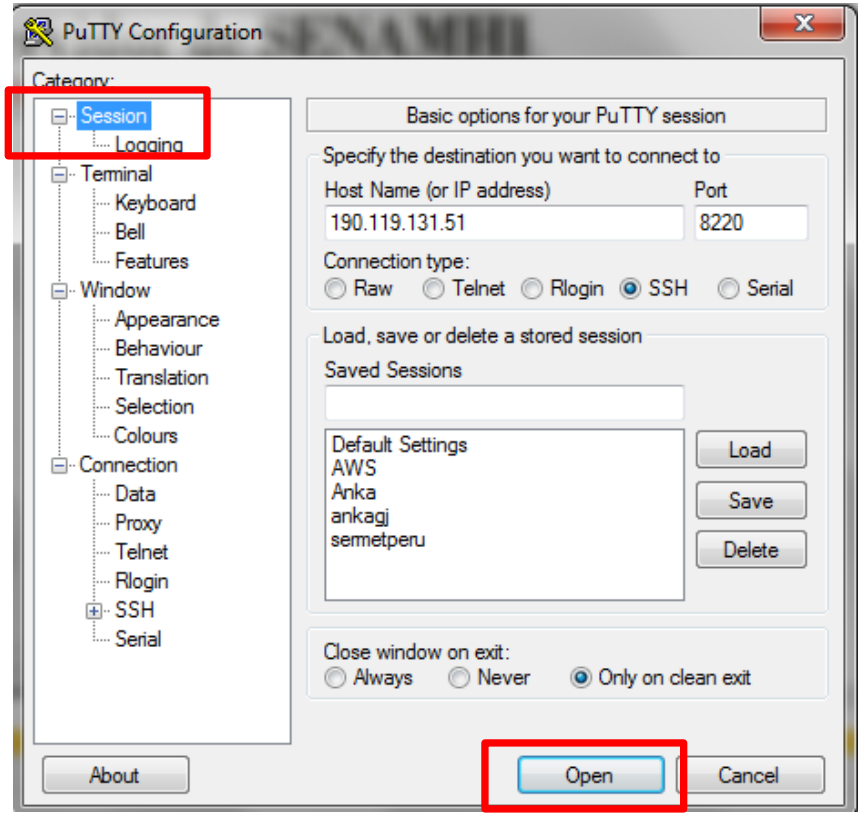
Acceso al servidor de SENAMHI



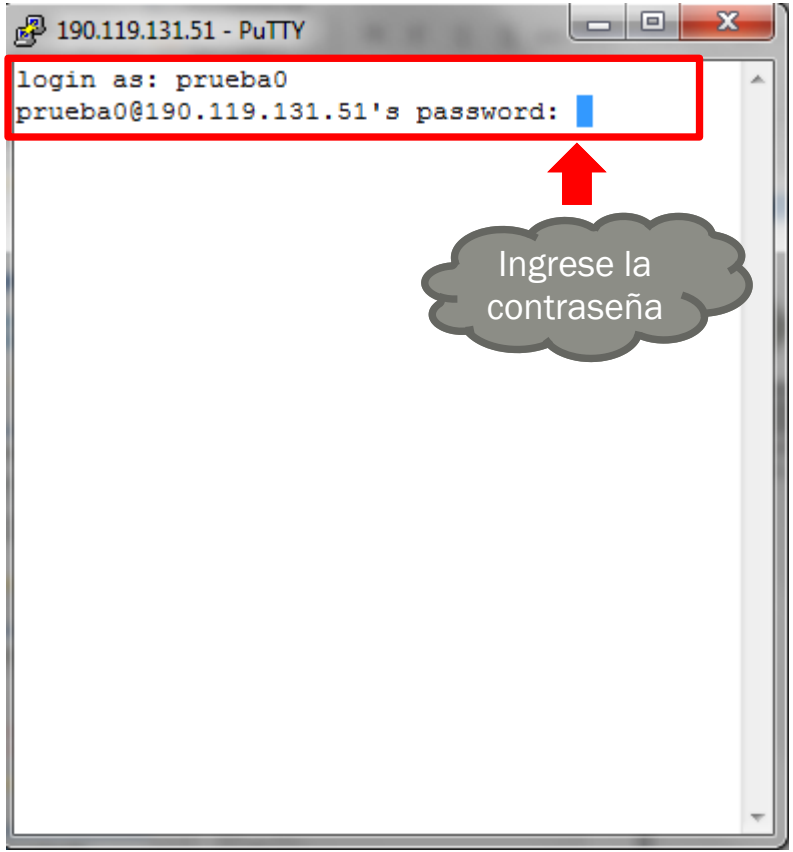


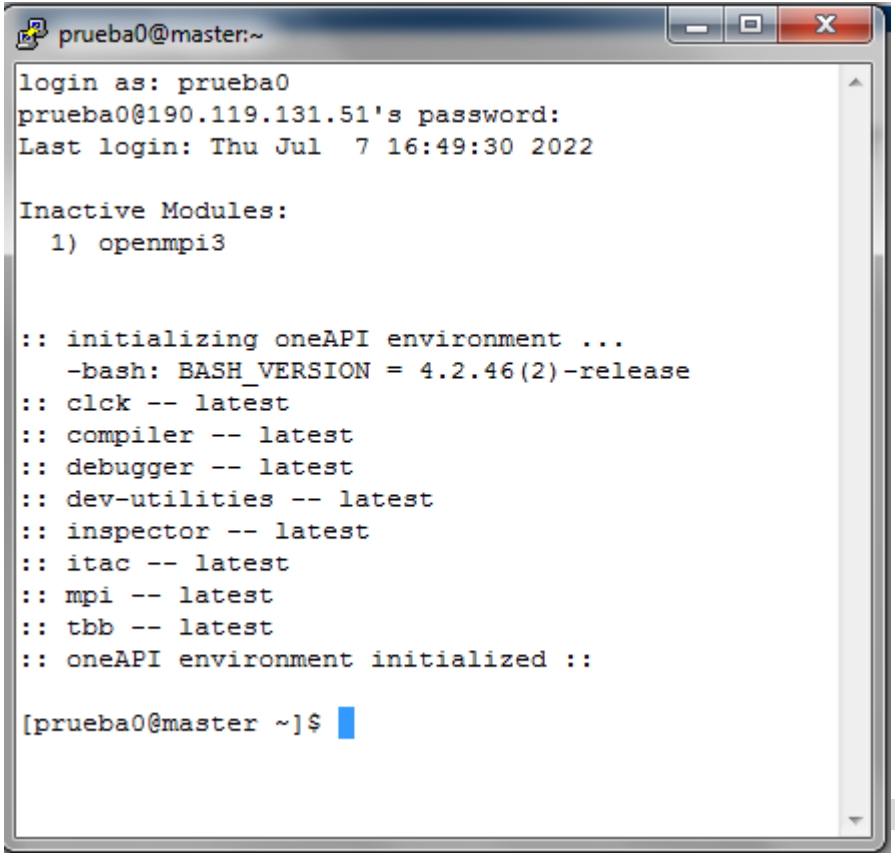
Acceso al servidor de SENAMHI





Acceso al servidor de SENAMHI





PERÚ Ministerio del Ambiente

Senamhi
SERVICIO NACIONAL DE METEOROLOGÍA E HIDROLOGÍA DEL PERÚ

Siempre con el pueblo

Comandos básicos de Linux

COMANDOS: Manejo de carpetas y archivos

pwd (Print Working Directory) : Muestra la ubicación de la carpeta actual

Sintaxis: pwd

Ejemplos:

>> pwd {Muestra la ubicación de la carpeta actual}

ls (LiSt) : Lista el contenido de una carpeta

Sintaxis: ls [opciones] [archivo o carpeta]

Ejemplos:

>> ls {Lista archivos en la carpeta actual}

>> ls -ltr .. {Lista archivos en la carpeta externa}

>> ls –help {Para ver opciones del comando.

cd (Change Directory) : Cambio de ubicación a otra carpeta

Sintaxis: cd *[carpeta]*

Ejemplos:

>> cd Documentos

COMANDOS: Manejo de carpetas y archivos

mkdir (MaKe DIRectory) : Construye una carpeta
Sintaxis: mkdir [nombre de la nueva carpeta]
Ejemplos:
>> mkdir EJEMPLOS {Construye la carpeta EJEMPLOS}
>> mkdir UNO {Construye la carpeta UNO}

rmdir (ReMove DIRectory) : Elimina o borra una carpeta vacía
Sintaxis: rmdir *[carpeta]*
Ejemplos:
>> cd UNO
>> cd ..
>> rmdir UNO {Elimina la carpeta vacía UNO}
>> cd EJEMPLOS

rm (ReMove) : Remueve o elimina un(os) archivo(s)
Sintaxis: rm [opciones] [archivos a borrar]

COMANDOS: Manejo de archivos

cp (CoPy) : Copia un(os) archivo(s).
Sintaxis: cp [opciones] [archivos a copiar] [archivo final]
rm (ReMove) : Remueve o elimina un(os) archivo(s)
Sintaxis: rm [opciones] [archivos a borrar]

touch : crea uno o varios archivos en blanco
Sintaxis: touch [opciones] [archivo]

Ejemplos:

>> pwd {Debemos estar en la carpeta EJEMPLOS}
>> touch .... (crear los días de la semana, ejem: LUNES.txt,....)
>> ls
>> cp LUNES.txt LUNES_01.txt {Copiar la información de LUNES.txt al nuevo fichero LUNES_01}
>> mv LUNES_01.txt DIA_01.txt {Cambiar de nombre al archivo LUNES_01.txt}
>> rm -f DIA_01.txt {Eliminar el archivo DIA_01.txt}

Ln (enlace directo): ln –s /ruta/inicio/file /ruta/final/

PERÚ Ministerio del Ambiente

Senamhi
SERVICIO NACIONAL DE METEOROLOGÍA E HIDROLOGÍA DEL PERÚ

Siempre con el pueblo

Comandos básicos GrADS

Visualizador grafico: GrADS

] grads (enter)

] reinit

] set display white

] c

] open file.ctl

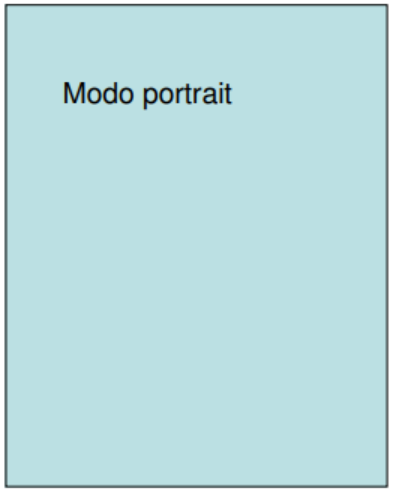
] set lat [lat1] [lat2]

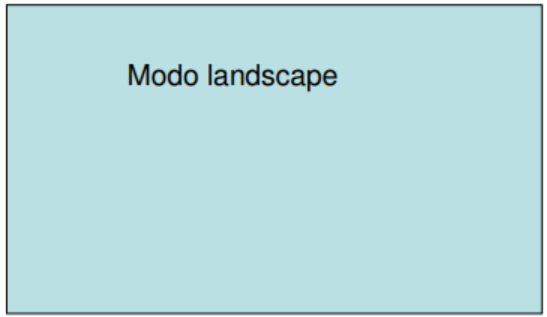
] set lon [lon1] [lon2]

] set gxout shaded

] d variable

En la ventana de comandos (terminal) GrADS pregunta si la ventana gráfica abrirá en modo apaisado (landscape) o vertical (portrait). El modo apaisado tiene una dimensión de 11x8.5 pulgadas y el modo vertical una de 8.5x11. El tamaño de la ventana gráfica puede ser modificado a posteriori (solo en Linux).





El modo a seleccionar depende de lo que se quiera graficar.

GRACIAS POR SU ATENCIÓN