



REPÚBLICA DEL ECUADOR

# Escuela Politécnica Nacional

"SCIENTIA HOMINIS SALUS"

La versión digital de esta tesis está protegida por la Ley de Derechos de Autor del Ecuador.

Los derechos de autor han sido entregados a la "ESCUELA POLITÉCNICA NACIONAL" bajo el libre consentimiento del (los) autor(es).

Al consultar esta tesis deberá acatar con las disposiciones de la Ley y las siguientes condiciones de uso:

- Cualquier uso que haga de estos documentos o imágenes deben ser sólo para efectos de investigación o estudio académico, y usted no puede ponerlos a disposición de otra persona.
- Usted deberá reconocer el derecho del autor a ser identificado y citado como el autor de esta tesis.
- No se podrá obtener ningún beneficio comercial y las obras derivadas tienen que estar bajo los mismos términos de licencia que el trabajo original.

El Libre Acceso a la información, promueve el reconocimiento de la originalidad de las ideas de los demás, respetando las normas de presentación y de citación de autores con el fin de no incurrir en actos ilegítimos de copiar y hacer pasar como propias las creaciones de terceras personas.

***Respeto hacia sí mismo y hacia los demás.***

# **ESCUELA POLITÉCNICA NACIONAL**

**FACULTAD DE INGENIERÍA ELÉCTRICA Y  
ELECTRÓNICA**

**AUTOMATIZACIÓN DEL MODELO DE PREDICCIÓN  
METEOROLÓGICA WRF EMPLEANDO HERRAMIENTAS DE  
SOFTWARE LIBRE**

**PROYECTO PREVIO A LA OBTENCIÓN DEL TÍTULO DE INGENIERO EN  
ELECTRÓNICA Y REDES DE INFORMACIÓN**

**PAÚL FRANCISCO CÁRDENAS LUCAS**  
panchocardenas83@gmail.com

**DIRECTOR: DR. LUIS ENRIQUE MAFLA GALLEGOS**  
mafla@epn.edu.ec

**Quito, Junio 2013**

## **DECLARACION**

Yo, Paúl Francisco Cárdenas Lucas, declaro bajo juramento que el trabajo aquí descrito es de mi autoría; que no ha sido previamente presentada para ningún grado o calificación profesional; y, que he consultado las referencias bibliográficas que se incluyen en este documento.

A través de la presente declaración cedo mis derechos de propiedad intelectual correspondientes a este trabajo, a la Escuela Politécnica Nacional, según lo establecido por la Ley de Propiedad Intelectual, por su Reglamento y por la normatividad institucional vigente.

Paúl Francisco Cárdenas Lucas

## CERTIFICACIÓN

Certifico que el presente trabajo fue desarrollado por Paúl Francisco Cárdenas Lucas, bajo mi supervisión.

Dr. Enrique Mafla Gallegos  
DIRECTOR DE PROYECTO



## AGRADECIMIENTO

*Gracias mamá y papá por su apoyo constante e inagotable paciencia.*

*“Directo, conciso y preciso”*

*Gracias a la acertada guía del Dr. Enrique Mafla que llevó a la culminación de este  
proyecto.*

*Mi sincero agradecimiento a mis profesores politécnicos por compartir sus conocimientos  
en las aulas de esta institución única.*

*Gracias amigas y amigos con quienes compartí tantas experiencias dentro y fuera de la  
EPN.*

## DEDICATORIA

*“A Juanita, José, Mile y Sole”*

## TABLA DE CONTENIDOS

TABLA DE CONTENIDOS .....	v
ÍNDICE DE FIGURAS .....	vii
ÍNDICE DE TABLAS .....	viii
CAPÍTULO 1 .....	1
INTRODUCCIÓN .....	1
1.1 IDENTIFICACIÓN DEL PROBLEMA .....	1
1.1.1 COMPONENTES DEL MODELO <i>WRF</i> .....	1
1.2 OBJETIVOS.....	8
1.2.1 OBJETIVO GENERAL .....	8
1.2.2 OBJETIVOS ESPECÍFICOS .....	8
1.3 ALCANCE.....	9
1.4 METODOLOGÍA PARA EL DESARROLLO DEL PROYECTO.....	9
1.5 ANÁLISIS DE REQUERIMIENTOS DE AUTOMATIZACIÓN .....	11
1.5.1 REQUERIMIENTOS DE COMPUTACIÓN.....	11
1.5.2 REQUERIMIENTOS DE DATOS DE ENTRADA .....	13
1.5.3 REQUERIMIENTOS DE ALMACENAMIENTO .....	14
1.5.4 REQUERIMIENTOS PARA PUBLICACIÓN DE RESULTADOS.....	15
CAPÍTULO 2 .....	17
AUTOMATIZACIÓN .....	17
2.1 ESTRUCTURA DE DIRECTORIOS.....	17
2.2 CONFIGURACIÓN DEL SERVIDOR.....	18
2.2.1 INSTALACIÓN DE PAQUETES BÁSICOS PARA <i>WRF</i> .....	18
2.2.2 INSTALACIÓN Y COMPILACIÓN DEL CÓDIGO FUENTE <i>WRF</i> .....	21
2.2.3 CONFIGURACIÓN DE SERVIDOR <i>WEB</i> .....	25
2.2.4 CONFIGURACIÓN DE SERVIDOR <i>FTP</i> .....	26
2.3 DISEÑO .....	27
2.3.1 DESCARGA DE DATOS.....	27
2.3.2 INGRESO DE DATOS A <i>WPS</i> .....	29
2.3.3 INGRESO DE DATOS AL BLOQUE DE PREDICCIÓN <i>WRF</i> .....	30
2.3.4 GRÁFICOS Y PUBLICACIÓN.....	31
2.4 <i>SCRIPTS</i> DE AUTOMATIZACIÓN .....	31
2.4.1 <i>SCRIPT</i> GENERAL.....	32
CAPÍTULO 3 .....	34

VALIDACIÓN Y PUBLICACIÓN .....	34
3.1 CASO IDEAL .....	36
3.2 CASO REAL .....	41
3.3 MODELO AUTOMATIZADO .....	52
CAPÍTULO 4 .....	62
CONCLUSIONES Y RECOMENDACIONES .....	62
4.1 CONCLUSIONES .....	62
4.2 RECOMENDACIONES .....	64
CAPÍTULO 5 .....	66
BIBLIOGRAFÍA Y ANEXOS .....	66
5.1 BIBLIOGRAFÍA .....	66
5.2 ANEXOS .....	67
ANEXO A. ARCHIVO DE CONFIGURACIÓN: <i>namelist.wps</i> .....	67
ANEXO B. ARCHIVO DE CONFIGURACIÓN: <i>namelist.input</i> .....	68
ANEXO C. SCRIPT: GENERAL.SH .....	69
ANEXO D. SCRIPT: 1_DESCARGA-DATOS.SH .....	71
ANEXO E. SCRIPT: 2_CAMBIAR_NAMELISTWPS_AS.SH .....	72
ANEXO F: 3_CORRER_PROGRAMASWPS_AS.SH .....	73
ANEXO G. SCRIPT: 4_CAMBIA_NAMELISTINPUT_AS.SH .....	74
ANEXO H. SCRIPT: 5_CORRER_PROGRAMASWRF_AS.SH .....	74
ANEXO I. SCRIPT: 6_GRAFICAR_AS.SH .....	75
ANEXO J. SCRIPT: 1_CAMBIAR_NAMELISTWPS_EC.SH .....	76
ANEXO K: 2_CORRER_PROGRAMASWPS_EC.SH .....	77
ANEXO L. SCRIPT: 3_CAMBIA_NAMELISTINPUT_EC.SH .....	77
ANEXO M. SCRIPT: 4_CORRER_PROGRAMASWRF_EC.SH .....	78
ANEXO N. SCRIPT: 5_GRAFICAR_EC.SH .....	78
ANEXO O. SCRIPT: 1_CAMBIAR_NAMELISTWPS_PI.SH .....	79
ANEXO P: 2_CORRER_PROGRAMASWPS_PI.SH .....	80
ANEXO Q. SCRIPT: 3_CAMBIA_NAMELISTINPUT_PI.SH .....	81
ANEXO R. SCRIPT: 4_CORRER_PROGRAMASWRF_PI.SH .....	81
ANEXO S. SCRIPT: 5_GRAFICAR_PI.SH .....	82
ANEXO T. GRÁFICAS DE SIMULACIÓN CASO IDEAL .....	83
ANEXO U. GRÁFICAS DE SIMULACIÓN PARA AMÉRICA DEL SUR. ....	86
ANEXO V. ARCHIVO <i>index.html</i> CORRESPONDIENTE AL SITIO WEB... ..	127

## ÍNDICE DE FIGURAS

Fig. 1.1 Diagrama de flujo del sistema de modelado WRF .....	2
Fig. 1.2 Ejemplo sencillo del modelo de datos <i>netCDF</i> .....	7
Fig. 2.1. Flujo de datos entre los programas de WPS .....	29
Fig. 3.1. Archivo de configuración <i>wrf_BWave.ncl</i> .....	40
Fig. 3.2. Archivos ejecutables y archivos de salida generados para el caso ideal. .....	40
Fig. 3.3. Opciones de ejecución de <i>WPS</i> .....	50
Fig. 3.4. Proceso de <i>geogrid.exe</i> y mensaje de ejecución exitosa .....	50
Fig. 3.5. Contenido del directorio <i>WPS</i> luego de descifrado y enlazado de archivos <i>FNL</i> .....	50
Fig. 3.6. Contenido del directorio <i>em_real</i> luego de enlazado de archivos <i>met_em</i> .....	50
Fig. 3.7. Mensaje de correcta ejecución de <i>real.exe</i> .....	50
Fig. 3.8. Mensaje de correcta ejecución de <i>wrf.exe</i> .....	50
Fig. 3.9. Contenido del directorio <i>WRF</i> luego de correcta ejecución de <i>wrf.exe</i> ..	50
Fig. 3.10. Ejecución exitosa del programa <i>geogrid.exe</i> .....	50
Fig. 3.11. Archivo <i>geo_em.d01.nc</i> creado con información del dominio .....	51
Fig. 3.12. Automatización. Programa <i>geogrid.exe</i> .....	53
Fig. 3.13. Automatización. Programa <i>metgrid.exe</i> .....	54
Fig. 3.14. Automatización. Programa <i>real.exe</i> y <i>wrf.exe</i> .....	54
Fig. 3.15. Automatización. Herramienta gráfica NCL .....	55
Fig. 3.16. Automatización. Página Web .....	55
Fig. 3.17. Automatización. Sitio ftp con el historial de archivos .....	56
Fig. 3.18. Imagen satelital NASA para el continente americano obtenida a las 14h45 del 2013-05-04 .....	57
Fig. 3.19. Imagen satelital para el continente americano obtenida a las 15h00 del 2013-05-04 .....	58
Fig. 3.20. Imagen de simulación obtenida a las 15h01 del 2013-05-04 .....	59
Fig. 3.21. Comparación entre dos imágenes satelitales y la imagen obtenida de la simulación .....	61

## ÍNDICE DE TABLAS

Tabla 1.1. Secciones del código GRIB2.....	3
Tabla 2.1. Nombres de directorios de datos meteorológicos de acuerdo a la hora de disponibilidad.....	28

## PRESENTACIÓN

Se automatizará el funcionamiento del modelo numérico de predicción meteorológica *Weather Research and Forecasting (WRF)* para lo cual se identificará el problema que existe con el uso del modelo *WRF*, luego se enumerarán los componentes del modelo y sus funciones, se analizarán requerimientos de automatización y finalmente se diseñarán e implementarán los procesos de entrada y salida de datos del modelo.

El uso del modelo *WRF* está ampliamente difundido dentro de la comunidad científica, pero al no existir una implementación automatizada del mismo, este proyecto se plantea automatizar su funcionamiento, debido a lo compleja que puede resultar su ejecución paso a paso.

La ejecución del modelo *WRF* implica la realización de varias actividades. Primero, la descarga de datos meteorológicos; segundo, la modificación de archivos de configuración; tercero, la ejecución de los módulos de pre-procesamiento; cuarto, la ejecución de los módulo de predicción; y quinto, el post-procesamiento de los datos obtenidos.

El proyecto planteado pretende evitar la realización de todas las tareas anteriormente descritas cada vez que se ejecuta el modelo. Para esto se diseñan e implementan procesos de entrada y salida de datos del modelo. Estos procesos se plasman en *scripts* de programación que automatizan el funcionamiento del modelo y presentan los resultados en un sitio web, facilitando así la experiencia del usuario.

El presente proyecto ayudará al lector a conocer el funcionamiento de los componentes del modelo, y a partir de eso comprender de mejor manera el proceso de automatización.

## RESUMEN

El presente proyecto busca automatizar el funcionamiento del modelo numérico de predicción meteorológica *Weather Research and Forecasting (WRF)*. El proyecto está compuesto de tres partes: el análisis de requerimientos para la automatización del modelo; el diseño e implementación de procesos de automatización; y finalmente la realización de pruebas para verificar el funcionamiento del modelo. A continuación se describe brevemente lo realizado en cada capítulo.

El capítulo 1 contiene el marco teórico respecto a los componentes del modelo *WRF*. Consta además la metodología a seguir para el desarrollo del proyecto y el análisis de los requerimientos de automatización.

Dentro del capítulo 2 se detallan el diseño y la implementación del modelo *WRF* automatizado. Se diseñan e implementan los procesos de entrada y salida de datos del modelo. La implementación comprende la instalación del servidor *WRF*, la programación de *scripts* para la automatización y la publicación de resultados.

En el capítulo 3 se presentan las pruebas para verificar que se cumplan los requerimientos expuestos en la sección de análisis. En primer lugar se comprueba el correcto funcionamiento del modelo tanto para uso manual como para uso automatizado; y en segundo lugar se presentan el desarrollo de las pruebas, los resultados de las mismas y los respectivos análisis para corroborar el cumplimiento del alcance del proyecto.

En el capítulo 4 se incluyen las conclusiones y recomendaciones luego de la terminación del proyecto. Por último la bibliografía utilizada y los anexos constarán en el capítulo 5.



# CAPÍTULO 1

## INTRODUCCIÓN

Este capítulo define el proyecto “Automatización del modelo numérico de predicción meteorológica *Weather Research and Forecasting-WRF* empleando herramientas de software libre”. Para la realización de este proyecto el capítulo presenta el problema que existe con el uso del modelo *WRF*. Además en el marco teórico se enumera los componentes del modelo y sus funciones. Posteriormente se describe la metodología a seguir para el desarrollo del proyecto. Para finalizar se realiza el análisis de requerimientos de automatización.

### 1.1 IDENTIFICACIÓN DEL PROBLEMA

La ejecución del modelo *WRF* implica la realización de varias actividades. Primero, la descarga de datos meteorológicos; segundo, la modificación de archivos de configuración; tercero, la ejecución de los módulos de pre-procesamiento; cuarto, la ejecución de los módulos de predicción; y quinto, el post-procesamiento de los datos obtenidos. Todas estas tareas deben ejecutarse manualmente cada vez que se necesite realizar una nueva simulación.

#### 1.1.1 COMPONENTES DEL MODELO *WRF*<sup>[1][2][3]</sup>

En esta sección se identifican los principales componentes del modelo y se enumeran sus respectivas funciones. La fig. 1.1 muestra el diagrama de flujo para el modelo *WRF* Versión 3 donde se aprecia la fuente externa de datos, el bloque de pre-procesamiento, el bloque de predicción, la modificación de archivos de configuración y el bloque de post-procesamiento y visualización.

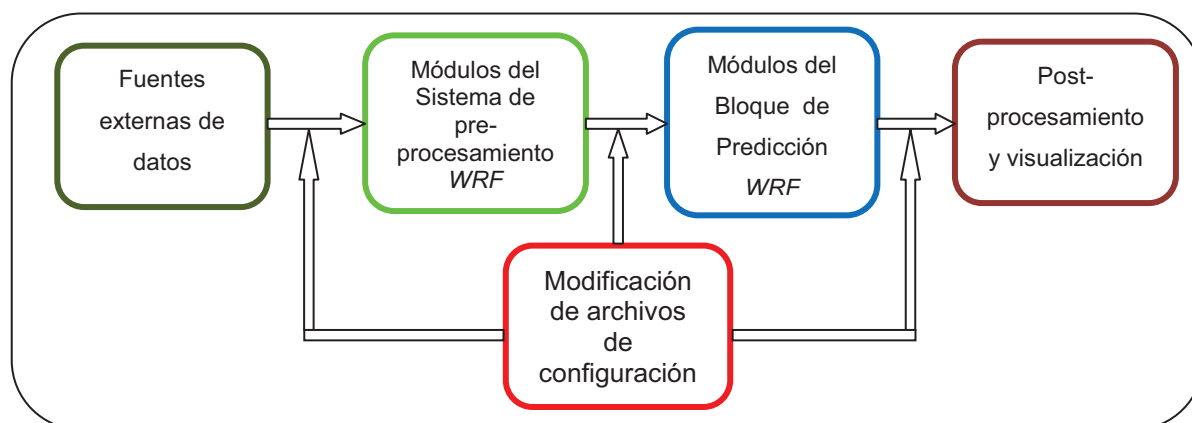


Fig. 1.1 Diagrama de flujo del sistema de modelado WRF.<sup>1</sup>

### 1.1.1.1 Fuentes externas de datos

Se emplean dos fuentes de datos principales para el funcionamiento del modelo WRF: datos geográficos estáticos y datos meteorológicos dinámicos.

Los datos geográficos estáticos contienen información necesaria para el modelo WRF como latitud, longitud y factores de escala de mapa de la región geográfica sobre la cual se van a realizar las predicciones. Estos datos no varían con el tiempo, por lo que necesitan ser descargados una sola vez.

Los datos meteorológicos dinámicos contienen campos de información meteorológica que provienen de otro modelo regional o global, por ejemplo del modelo “GFS”<sup>2</sup>. Estos datos son utilizados por WRF para establecer las condiciones iniciales. Pueden aparecer en varios formatos, pero el más empleado es el GRIB2 por la facilidad que presenta al codificar y añadir nuevos campos climatológicos y de pronóstico que necesitan del uso de valores estadísticos.

*General Regularly-distributed Information In Binary - GRIB* (por sus siglas en inglés), es el nombre de una forma de representación de datos para información general regularmente distribuida en binario. Los datos codificados en GRIB2

<sup>1</sup> Fuente: Chapter 1- Overview, UCAR.

<sup>2</sup> Global Forecast System – GFS: <ftp.ncep.noaa.gov/pub/data/nccf/com/gfs/prod/gfs.YYYYMMDDHH>.

consisten de un flujo continuo de bits hecho de una secuencia de octetos (1 octeto = 8 bits). A continuación en la tabla 1.1 se muestran los octetos de un mensaje *GRIB2* agrupados en secciones:

Número de Sección	Nombre de la sección	Contenido de la Sección
0	Indicador	"GRIB", Disciplina, número de la edición <i>GRIB</i> , longitud del mensaje
1	Identificación	Longitud de la sección, número de la sección, características que aplican a todos los datos procesados en el mensaje <i>GRIB</i>
2	Uso Local (opcional)	Longitud de la sección, número de la sección, ítems adicionales para uso local
3	Definición de Grilla	Longitud de la sección, número de la sección, definición de superficie de grilla y geometría de los valores de los datos dentro de la grilla
4	Definición de Producto	Longitud de la sección, número de la sección, descripción de la naturaleza de los datos
5	Representación de Datos	Longitud de la sección, número de la sección, descripción de cómo son representados los valores de los datos
6	Bit-map	Longitud de la sección, número de la sección, indicación de presencia o ausencia de datos en cada uno de los puntos de la grilla
7	Datos	Longitud de la sección, número de la sección, valores de los datos
8	Final	"7777"

Tabla 1.1. Secciones del código GRIB2.<sup>3</sup>

### 1.1.1.2 Módulos del Sistema de pre-procesamiento *WRF*

El sistema de pre-procesamiento *WRF* – *WRF Pre-processing System* (*WPS* por sus siglas en inglés) es utilizado para simulaciones de datos reales. *WPS* utiliza un archivo de configuración: *namelist.wps* el mismo que está compuesto de tres módulos: *geogrid*, *ungrib*, *metgrid*.

El archivo *namelist.wps* posee información sobre parámetros necesarios para el trabajo de los módulos. Esta información está dividida en varios registros: un registro compartido y registros separados para cada uno de los módulos (ver Anexo A).

<sup>3</sup> Fuente: <http://www.wmo.int/pages/prog/www/DPS/FM92-GRIB2-11-2003.pdf>

#### 1.1.1.2.1 Módulo *geogrid*

El módulo *geogrid* define dominios de simulación. Los dominios de simulación son definidos usando información especificada por el usuario en la sección “*geogrid*” del archivo de configuración *namelist.wps*. La información requerida es latitud, longitud y escala del mapa de la región geográfica sobre la cual se van a realizar las predicciones.

Además el módulo *geogrid* interpola datos geográficos (categorías de suelo, altura de terreno, temperatura del suelo, etc) en el dominio de simulación. Estos datos son obtenidos de un conjunto de datos globales, son invariantes en el tiempo y por esta razón se descargan una sola vez.

#### 1.1.1.2.2 Módulo *ungrib*

El módulo *ungrib* decodifica el conjunto de datos *GRIB2* en un formato simple que pueda ser entendido por el módulo *metgrid*. Además, el formato *GRIB2* emplea varios códigos para identificar variables y niveles en el archivo. El módulo *ungrib* utiliza tablas de estos códigos para definir cuáles campos extraer del archivo *GRIB2* y así escribirlos en formato intermedio.

#### 1.1.1.2.3 Módulo *metgrid*

El módulo *metgrid* interpola los datos meteorológicos en formato intermedio que son extraídos por el módulo *ungrib* dentro de los dominios de simulación. Así de esta forma, la salida interpolada de *metgrid* puede ser asimilada por el módulo *WRF*.

### 1.1.1.3. Módulos del Bloque de Predicción *WRF*

El bloque de predicción *WRF* utiliza un archivo de configuración: *namelist.input* y está compuesto de los módulos: *real*, y *wrf*. Cada uno de los módulos lee información del archivo *namelist.input*. Esta información está dividida en cuatro

registros. El primer registro de tiempo y control especifica las fechas de inicio y finalización de la simulación. El segundo registro brinda información sobre latitud, longitud y número de dominios de simulación. El tercer y cuarto registro sirve para definir opciones físicas y dinámicas (ver Anexo C).

#### *1.1.1.3.1 Módulo real*

El módulo *real* interpola verticalmente los datos generados por *WPS*. El modelo *WRF* posee coordenadas verticales que siguen la topografía con una superficie de presión constante. El módulo *real* se encarga de interpolar los datos a las coordenadas verticales del modelo.

Además, el módulo *real* crea archivos de condiciones iniciales y de frontera. Las condiciones de frontera son las condiciones meteorológicas de los bordes del área de ejecución del modelo representadas en forma matemática.

#### *1.1.1.3.2 Módulo wrf*

El módulo *wrf* realiza la simulación numérica con tiempos de ejecución establecidos en el archivo de configuración *namelist.input*. Los datos obtenidos de la simulación numérica se obtienen en formato *netCDF*.

*Network Common Data Format - NetCDF* es un conjunto de formatos de datos, interfaces de programación, y librerías de *software* que ayudan a la lectura y escritura de archivos de datos científicos. El modelo de datos *netCDF* consiste de variables, dimensiones, y atributos, los cuales tienen un nombre y un número de ID para poder ser identificados.

Las variables son arreglos N-dimensionales de datos. Una variable tiene un nombre, un tipo de datos, y una forma descrita por su lista de dimensiones especificada cuando la variable es creada. Las variables en archivos *netCDF* pueden ser de seis tipos (caracter, *byte*, *short*, entero, flotante, doble).

Las dimensiones describen los ejes de los arreglos de datos. Una dimensión tiene un nombre y una longitud, y puede ser usada para representar dimensiones físicas reales.

Los atributos son variables o archivos con pequeñas notas o datos suplementarios. Estos datos son identificados por el nombre (o ID) de esa variable, junto con el nombre del atributo. Los atributos son siempre valores escalares o arreglos de 1 dimensión, los cuales pueden estar asociados con una variable o el archivo completo. Aunque no existe un límite establecido, se espera del usuario mantener los atributos pequeños.

En meteorología, *netCDF* considera la existencia de varias cantidades físicas (presión, temperatura) localizadas en puntos en una latitud, longitud, nivel vertical y tiempo particular. La información de los ejes (latitud, longitud, nivel, y tiempo) son almacenadas como dimensiones *netCDF*. Las cantidades físicas (presión, temperatura) son almacenadas como variables *netCDF*.

Los bits adicionales de datos son almacenados como atributos *netCDF*. Los atributos son siempre valores simples o arreglos de una dimensión. A continuación en la figura 1.2 se empleará un ejemplo sencillo para ilustrar el concepto del modelo de datos *netCDF*.

```
netcdf example_1 { // example of CDL notation for a netCDF dataset

dimensions: // dimension names and lengths are declared first
    lat = 5, lon = 10, level = 4, time = unlimited;

variables: // variable types, names, shapes, attributes
    float temp(time,level,lat,lon);
        temp:long_name = "temperature";
        temp:units = "celsius";
    float rh(time,lat,lon);
        rh:long_name = "relative humidity";
        rh:valid_range = 0.0, 1.0; // min and max
    int lat(lat), lon(lon), level(level);
        lat:units = "degrees_north";
        lon:units = "degrees_east";
        level:units = "millibars";
    short time(time);
        time:units = "hours since 1996-1-1";
// global attributes
    :source = "Fictional Model Output";
```

```

data: // optional data assignments
      level = 1000, 850, 700, 500;
      lat = 20, 30, 40, 50, 60;
      lon = -160, -140, -118, -96, -84, -52, -45, -35, -25, -15;
      time = 12;
      rh = .5, .2, .4, .2, .3, .2, .4, .5, .6, .7,
           .1, .3, .1, .1, .1, .1, .5, .7, .8, .8,
           .1, .2, .2, .2, .2, .5, .7, .8, .9, .9,
           .1, .2, .3, .3, .3, .3, .7, .8, .9, .9,
           0, .1, .2, .4, .4, .4, .4, .7, .9, .9;
}

```

Fig. 1.2. Ejemplo sencillo del modelo de datos netCDF.<sup>4</sup>

En el presente ejemplo se consideran cuatro dimensiones que están bajo la directiva *dimensions*. Las variables se encuentran bajo la directiva *variables* y finalmente los atributos se encuentran bajo la directiva *data*.

#### 1.1.1.4 Post-procesamiento y visualización

Los datos del modelo se obtienen en formato *netCDF* y existen varios programas empleados para interpretar los resultados obtenidos por el modelo *WRF*. De toda la variedad existente se escogió el *National Center for Atmospheric Research Command Language* (*NCL* por sus siglas en inglés) que es un lenguaje diseñado especialmente para análisis de datos científicos y visualización que soporta formatos *netCDF*, *GRIB1*, *GRIB2* y binarios. *NCL* posee varias características comunes a los lenguajes de programación modernos, incluyendo tipos, variables, operadores, expresiones, lazos, funciones y procedimientos. En adición a estas características, *NCL* posee funciones que no se encuentran en otro tipo de lenguajes, incluyendo la configuración de las visualizaciones, el importado de datos de una variedad de formatos y soporte para operaciones de arreglos. La razón de trabajar con *NCL* radica en que es una herramienta de libre acceso, de sencilla instalación y con varios *scripts* de apoyo para la visualización de los datos obtenidos del modelo.

<sup>4</sup> Fuente: <http://www.unidata.ucar.edu/software/netcdf/docs/netcdf.pdf>

Como se ha explicado en los apartados anteriores, la ejecución del modelo *WRF* implica la realización de varias actividades. Primero, la descarga de datos meteorológicos cuya estructura es compleja; segundo, la modificación de archivos de configuración; tercero, la ejecución de los módulos de pre-procesamiento; cuarto, la ejecución de los módulo de predicción; y quinto, el post-procesamiento de los datos obtenidos en formato *netCDF*. El usuario debe realizar manualmente todas estas actividades cada vez que se necesite realizar una nueva simulación. Para facilitar el trabajo del usuario este proyecto plantea el funcionamiento automático del modelo *WRF*. Los objetivos y el alcance de este proyecto se detallan en los apartados siguientes.

## **1.2 OBJETIVOS**

### **1.2.1 OBJETIVO GENERAL**

Automatizar el funcionamiento del modelo numérico de predicción meteorológica *WRF*, con el fin de obtener predicciones diarias en tres dominios; América del Sur, Ecuador y Pichincha.

### **1.2.2 OBJETIVOS ESPECÍFICOS**

- Analizar los requerimientos de automatización del modelo numérico *WRF*.
- Descargar los datos iniciales desde una fuente externa.
- Ingresar automáticamente los datos a los módulos del sistema de pre-procesamiento para luego obtener los resultados gracias a las herramientas de post-procesamiento y visualización.
- Comprobar el funcionamiento del sistema automatizado.
- Validar los resultados obtenidos mediante la comparación de los resultados con imágenes de satélite.
- Facilitar el acceso a los resultados, por parte del público en general mediante la publicación automática de las predicciones obtenidas en un sitio web.



### 1.3 ALCANCE

El presente proyecto comprende una fase descriptiva del funcionamiento del modelo numérico *WRF* y la configuración e instalación de un sistema operativo *open source* donde se realizará el proceso de automatización.

La fase de automatización engloba:

- El diseño de procesos de automatización.
- La instalación del *software* del sistema de modelación *WRF*, para lo cual se deben tener instalados correctamente los lenguajes de *script* y compiladores, fijar variables de ambiente UNIX para el adecuado funcionamiento de *WRF* y ejecutar el código *WRF*.
- Un proceso previo de simulaciones de entrenamiento del modelo numérico *WRF*.
- La programación de *scripts* dedicados a la automatización.
- El post-procesamiento de datos.

En la parte de validación se compararán los resultados de las simulaciones con imágenes satelitales y finalmente se realizará la publicación por medio de la web de los datos obtenidos.

### 1.4 METODOLOGÍA PARA EL DESARROLLO DEL PROYECTO

Para el desarrollo del presente proyecto se consideran algunas etapas que serán descritas brevemente en esta sección.

El análisis de requerimientos de automatización. De acuerdo a la documentación disponible se realizará el análisis de los requerimientos de automatización de los procesos de entrada y salida de datos del modelo *WRF*. Los documentos disponibles son:

- Tutorial sobre *WRF*:

<http://www.mmm.ucar.edu/wrf/users/supports/tutorial.html>

- Notas técnicas sobre *WRF*:

<http://www.mmm.ucar.edu/wrf/users/pub-doc.html>

El diseño de la automatización. El diseño del sistema se basa en el análisis obtenido de la sección 1.5. En el diseño se describen los procesos de entrada y salida de datos del modelo y su respectiva publicación en el sitio web.

La implementación de la automatización. Para la implementación de la automatización se emplea un entorno de máquina virtual. El motor de máquina virtual usado para el fin mencionado es VMware®. La utilización del entorno de máquina virtual se justifica debido a la facilidad de mantener varias copias del mismo sistema operativo base en caso de que existan problemas durante la instalación y configuración de paquetes.

Luego de configurar los parámetros de *hardware* de la máquina virtual se procede a: la instalación del sistema operativo y, de los paquetes necesarios para el funcionamiento del modelo; configuración del almacenamiento, configuración del servidor *Web* y *FTP*; finalmente programación de *scripts* en base al análisis de requerimientos para los procesos de entrada y salida de datos.

El sistema operativo escogido para el desarrollo de este proyecto es *Ubuntu Desktop* en su versión 11.4. Este sistema operativo *Open Source* posee los compiladores, los intérpretes y los utilitarios necesarios para el correcto funcionamiento del modelo.

Finalmente la validación de resultados. Esta sección presenta las pruebas para verificar que se cumplan los requerimientos de automatización. Para ello se comprobará el correcto funcionamiento del modelo con simulaciones de entrenamiento o ideales.

Además se realizan pruebas para corroborar el cumplimiento del alcance del proyecto. Esto es verificar los resultados de las simulaciones con imágenes satelitales y publicar los mismos en un sitio web para acceso del público. La realización de las pruebas, así como sus resultados, se detallan en el capítulo 3.

## 1.5 ANÁLISIS DE REQUERIMIENTOS DE AUTOMATIZACIÓN

En esta sección se analizan los requerimientos para la automatización del modelo *WRF*.

### 1.5.1 REQUERIMIENTOS DE COMPUTACIÓN <sup>[4]</sup> <sup>[5]</sup>

Las leyes físicas del movimiento y conservación de la energía rigen el desarrollo de las condiciones de la atmósfera. Estas leyes pueden expresarse por medio de una serie de ecuaciones matemáticas complejas que son imposibles de resolver con exactitud. *WRF* emplea métodos numéricos para obtener soluciones aproximadas. Si se conoce el estado inicial de la atmósfera es posible resolver la ecuación para un momento en el futuro.

*WRF* se ejecuta a distintas resoluciones horizontales y existe una relación directa entre la resolución horizontal y el tamaño de las características meteorológicas que se puedan simular.

*WRF* es además un modelo de malla que resuelve las ecuaciones de pronóstico en puntos de malla separados por distancias regulares. Las variables de pronóstico se especifican en un conjunto de puntos de malla. Normalmente, se necesitan al menos cinco puntos de malla para definir una característica. Al aumentar la resolución, los recursos informáticos necesarios también aumentan, porque *WRF* debe calcular los valores para un mayor número de puntos de malla.

*WRF* es un modelo intensivo en *CPU* por la gran cantidad de cálculos que debe realizar para establecer los pronósticos. Los procesos de entrada y salida de

datos del modelo apenas consumen recursos de *CPU*. Para facilidad del usuario, el modelo puede ser construido para correr sobre una máquina de procesador sencillo, sobre una máquina con memoria compartida (que use la *API OpenMPI*), sobre una máquina con memoria distribuida (con las librerías apropiadas *MPI*), o sobre un *cluster* distribuido (utilizando *OpenMP* y *MPI*).

#### **1.5.1.1 Código fuente**

El primer requerimiento indispensable son los códigos fuente de *WRF* y *WPS* (*WRFVx.x.x.TAR.gz* y *WPSVx.x.x.TAR.gz*). Éstos pueden obtenerse desde el área de descargas del sitio web para usuarios *WRF* en el siguiente enlace:  
[http://www.mmm.ucar.edu/wrf/users/download/get\\_sources.html](http://www.mmm.ucar.edu/wrf/users/download/get_sources.html).

Estos dos códigos fuente son la base para el funcionamiento del modelo.

#### **1.5.1.2 Compiladores y lenguaje de *script***

El equipo donde se instalen los componentes del modelo *WRF* debe cumplir mínimo con lo siguiente:

Compiladores:

- Compilador Fortran 90/95.
- Compilador C.

Intérpretes:

- Lenguaje Perl 5.04 o superior.
- Perl-TK (para la interfaz gráfica).

Shells

- Cshell
- Bourne Shell

Utilitarios

- Make
- Sed
- Awk

#### Librerías

- Librería *netCDF*.

## 1.5.2 REQUERIMIENTOS DE DATOS DE ENTRADA <sup>16]</sup>

### 1.5.2.1 Datos geográficos

El primer requerimiento de datos de entrada es el correspondiente al módulo *geogrid*. En este bloque se define el o los dominios de simulación y se interpola varios datos terrestres a las grillas del modelo.

Este bloque de datos comprimido pesa 429 MB y 11 GB descomprimido. Además, la ubicación de estos datos debe ser única y se tomará en cuenta para fijar variables de ambiente.

### 1.5.2.2 Datos meteorológicos

#### 1.5.2.2.1 Datos GFS

*Global Forecast System-GFS* son datos de tipo *GRIB2* y están disponibles en el siguiente enlace:

- <ftp.ncep.noaa.gov/pub/data/nccf/com/gfs/prod/gfs.YYYYMMDDHH>

Son datos disponibles sólo en tiempo real. La desventaja es el peso de los mismos que está entre 45 y 50 GB, pero son los que brindan resultados más cercanos a la realidad.

#### 1.5.2.2.2 Datos FNL

FNL (*Final Analyses*) son datos globales que están disponibles cada 6 horas. Poseen información almacenada desde el 15 de Septiembre de 1999 en el siguiente enlace:

- <http://dss.ucar.edu/datasets/ds083.2>

Los usuarios registrados pueden escoger entre tres *scripts* para realizar la descarga de datos.

### 1.5.3 REQUERIMIENTOS DE ALMACENAMIENTO

Respecto al almacenamiento debemos considerar tres aspectos:

- Los datos geográficos para el módulo *geogrid*.
- Los datos meteorológicos para el módulo *ungrib*.
- Los datos de salida del modelo que estarán disponibles en forma de historial para el público en general.

#### 1.5.3.1 Datos de entrada

Los datos geográficos empleados en el módulo *geogrid* deben permanecer en un directorio único. Estos datos pesan 11GB ya que poseen información geográfica global que nos permite definir dominios de simulación. La ubicación de este directorio se emplea para fijar una variable de ambiente.

En el módulo *ungrib* se emplean los datos GFS que están disponibles sólo en tiempo real. La desventaja que presentan estos datos es su peso, que varía entre 45 y 50 GB.

Los pasos para el manejo de estos datos son:

- Descarga de datos.
- Lectura y decodificación.
- Borrado de datos luego de ser procesados.

Se procede finalmente al borrado de los datos *GFS* pues ocupan demasiado espacio y luego de ser empleados por el respectivo módulo dentro del modelo, no representan un aporte para la realización de las simulaciones.

#### **1.5.3.2 Datos de salida del modelo**

Los datos de salida con formato *.nc* que poseen la información de las simulaciones permanecerán dentro del directorio en el cual se generan. Tiene un peso aproximado de 1 GB cada uno. Son tres archivos creados en cada simulación diaria y que corresponden a los tres dominios: América del Sur, Ecuador y Pichincha. Después serán enlazados a otra ubicación dentro del mismo sistema y allí estarán accesibles para el público.

#### **1.5.4 REQUERIMIENTOS PARA PUBLICACIÓN DE RESULTADOS**

El alcance del proyecto establece que, los resultados finales de las simulaciones serán mostrados en un sitio web. Éste deberá presentar información concreta respecto al objetivo del proyecto. El sitio debe incluir:

- Un listado de los archivos formato *PDF* correspondientes a los tres dominios propuestos.
- Mantener un historial de los archivos de salida de formato *.nc*, correspondientes a la salida del modelo antes de ser graficados y que estarán disponibles libremente para investigaciones de fenómenos climáticos.

Para manejar un propio sitio web, se realizarán los siguientes pasos:

- Instalar y levantar el servicio de Apache2, el servidor web más utilizado en sistemas Linux.
- Editar el archivo *index.html* con un editor de páginas web de acuerdo a nuestros requerimientos.
- Desarrollar el portal web y probarlo en la red interna.
- Configurar el dispositivo de enrutamiento para que exista concordancia entre los datos del dominio registrado y la dirección IP interna del servidor web.
- Comprobar que el servidor puede ser accedido por el público en general.



## CAPÍTULO 2

### AUTOMATIZACIÓN

En este capítulo se especifica la estructura de directorios del servidor *WRF*; luego se detalla el proceso de instalación de paquetes básicos, la compilación del código fuente del modelo *WRF*, y la instalación de los servidores Web y FTP; después se presenta el diseño de los procesos de entrada y salida de datos tomando en cuenta el análisis de requerimientos del capítulo anterior. Finalmente se realiza un listado de los *scripts* que permiten el funcionamiento del modelo *WRF* automatizado.

#### 2.1 ESTRUCTURA DE DIRECTORIOS

En esta sección se especifica la ubicación de: los datos geográficos y meteorológicos, el código fuente de *WRF*, la librería *netCDF*, la herramienta gráfica *NCL*, y los *scripts* que permiten la automatización. La estructura que debe tener el directorio *home/* es la siguiente:

- */home/user/Desktop/wrf1/*. Este es el directorio raíz desde donde se ramifican los directorios del código *WRF*, los directorios de los datos geográficos y meteorológicos, y los directorios de los servidores Web y *FTP*.
- */home/user/Desktop/wrf1/ftpdata/*. Es el nuevo directorio home del usuario *FTP*.
- */home/user/Desktop/wrf1/Webserver/*. Directorio donde se encuentra la información del sitio web (*index.html*, archivos *PDF*).
- */home/user/Desktop/wrf1/WRF/*. Aquí se almacenarán los datos de entrada, geográficos y meteorológicos; el código fuente de *WRF* y *WPS*.
- */home/user/Desktop/wrf1/WRF/DATA/*. En este directorio se guardan los datos de entrada meteorológicos obtenidos desde *NCEP*.

- */home/user/Desktop/wrf1/WRF/geog/*. Almacena los datos de entrada estáticos geográficos necesarios para definir los dominios.
- */home/user/Desktop/wrf1/WRF/WPS/*. Guarda el código fuente de *WPS*, dominios y archivos intermedios.
- */home/user/Desktop/wrf1/WRF/WRFV3/*. Almacena el código fuente de *WRF*, y los archivos de salida de las simulaciones antes de ser graficados.
- */home/user/ncl/*. Aquí se encuentran los archivos binarios pre-compilados de la herramienta gráfica *NCL*.
- */home/user/netCDF/*. Almacena la librería externa *netCDF* para el funcionamiento de *WRF*.
- */home/user/scripts/*. Aquí se encuentra el *script* general de automatización del modelo.
- */home/user/scripts/scripts-americaelsur/*. *Scripts* correspondientes al dominio de América del Sur.
- */home/user/scripts/scripts-ecuador/*. *Scripts* correspondientes al dominio de Ecuador.
- */home/user/scripts/scripts-pichincha/*. *Scripts* correspondientes al dominio de Pichincha.

## 2.2 CONFIGURACIÓN DEL SERVIDOR

En esta sección se detalla la instalación de los paquetes de software requeridos para el correcto funcionamiento de *WRF*, luego se presenta el proceso de instalación y compilación del código fuente de *WRF*, y finalmente se describe la instalación y modificación de archivos de configuración de los servidores Web y *FTP*.

### 2.2.1 INSTALACIÓN DE PAQUETES BÁSICOS PARA *WRF*

En esta sección se presentan los comandos empleados para la instalación de los paquetes básicos para el funcionamiento de *WRF* que incluyen: compiladores, intérpretes, utilitarios, librería *netCDF* y herramienta gráfica *NCL*.

Para el compilador *Fortran* se ejecuta la instrucción:

```
~$ sudo apt-get install gfortran
```

Para el lenguaje *Perl-TK* se escribe el comando:

```
~$ sudo apt-get install perl-tk
```

Además de los paquetes recomendados en la documentación oficial, surgieron inconvenientes con el funcionamiento del modelo, por lo que fue necesaria la instalación de paquetes adicionales sugeridos en la página de *Zephyr*:

- <http://zephyr.ucd.ie/mediawiki/index.php/WRF>.

Para la instalación de los paquetes adicionales se ejecutan los siguientes comandos:

```
~$ sudo apt-get install autoconf
~$ sudo apt-get install bison
~$ sudo apt-get install libnetcdf4
~$ sudo apt-get install libnetcdf-dev
~$ sudo apt-get install libjasper1
~$ sudo apt-get install libjasper-dev
~$ sudo apt-get install libpng12
~$ sudo apt-get install libpng-dev
~$ sudo apt-get install zlib1g
~$ sudo apt-get install zlib1g-dev
~$ sudo apt-get install libx11-dev
~$ sudo apt-get install libf2c2
~$ sudo apt-get install libf2c2-dev
~$ sudo apt-get install curl
~$ sudo apt-get install libcurl4-gnutls-dev
~$ sudo apt-get install flex
~$ sudo apt-get install autoconf
~$ sudo apt-get install m4
```

```
~$ sudo apt-get install original-awk
```

Dentro del directorio *home*/se crea el directorio *netcdf*/. En esta ubicación se almacena el paquete *netcdf-4.1.2.tar.gz* descargado de la página de Unidata.

Se descomprime el archivo usando los siguientes comandos:

```
~$ gunzip netcdf-4.1.2.tar.gz
```

```
~$ tar -xf netcdf-4.1.2.tar
```

Esto crea el directorio: *netcdf-4.1.2*/. A continuación se compila *netCDF* con los comandos:

```
~$ cd netcdf-4.0.1
```

```
~$ ./configure
```

```
~$ make (construye las librerías netCDF)
```

```
~$ makecheck (prueba la construcción)
```

```
~$ makeinstall (instala las librerías – lib, bin, include, share)
```

Al final del proceso de compilación se obtiene un mensaje confirmando la exitosa instalación de *netCDF*.

Como siguiente paso se fija la variable de entorno *NETCDF*. Para esto se añade la siguiente línea en el archivo *~/.bashrc* especificando la ubicación de *netCDF*:

```
export NETCDF=/home/user/netcdf/
```

Para la realizar la interpretación gráfica de los resultados obtenidos del modelo es necesario instalar una herramienta gráfica. En este caso se ha optado por *NCL* por su facilidad al momento de instalar. El paquete pre-compilado se obtiene desde el sitio de descarga de *NCAR CommandLanguage*:

- <http://www.ncl.ucar.edu/Download/>.

Se crea el directorio *ncl/* dentro de *home/*. Aquí se almacena el archivo *ncl\_ncarg-6.0.0.Linux\_Debian\_i686\_nodap\_gcc445.tar.gz*.

Se descomprime el archivo con los comandos:

```
~$ gunzip ncl_ncarg-
6.0.0.Linux_Debian_i686_nodap_gcc445.tar.gz
~$ tar -xf ncl_ncarg-
6.0.0.Linux_Debian_i686_nodap_gcc445.tar
```

Esto extrae los directorios *bin/*, *include/* y *lib/*.

Ahora es necesario indicarle al sistema la ubicación de *NCL*. Para ello en el archivo *~/.bashrc* se fijan tres variables de ambiente: *NCARG\_ROOT*, *PATH* y *MANPATH*:

```
export NCARG_ROOT=/home/user/ncl/
export PATH=/home/user/ncl/bin:$PATH
export MANPATH=/home/user/ncl/man:$MANPATH
```

## 2.2.2 INSTALACIÓN Y COMPILACIÓN DEL CÓDIGO FUENTE *WRF*

En la presente sección se detalla la ubicación de los datos geográficos, se describe el proceso de configuración y compilación del código *WRF* y se presentan los archivos creados luego de la compilación, y finalmente se describe el proceso de configuración y compilación del código *WPS* y se presentan los archivos creados luego de la compilación.

### 2.2.2.1 Datos geográficos

Se descargan los datos geográficos desde el sitio *web* de usuarios *WRF*<sup>5</sup>. Estos datos se almacenan en el directorio: */home/user/Desktop/wrf1/WRF/geog/*. En esa

---

<sup>5</sup> [http://www.mmm.ucar.edu/wrf/OnLineTutorial/Basics/GEOGRID/ter\\_data.htm](http://www.mmm.ucar.edu/wrf/OnLineTutorial/Basics/GEOGRID/ter_data.htm)

ubicación se descomprime el archivo *geog\_v3.1.tar.gz* empleando los siguientes comandos:

```
~$ gunzip geog_v3.1.TAR.gz
~$ tar -xf geog_v3.1.TAR
```

### 2.2.2.2 Configuración y compilación del código de *WRF*

Se descarga el código fuente para *WRF*. Este código se almacena en el directorio de trabajo: */home/user/Desktop/wrf1/WRF/*. El archivo descargado se descomprime usando los siguientes comandos:

```
~$ gunzip WRFV3.2.1.TAR.gz
~$ tar -xf WRFV3.2.1.TAR
```

Esto crea el directorio: *WRFV3/*. Es necesario ubicarse en este directorio creado con el comando:

```
~$ cd WRFV3
```

Dentro de este directorio, se encuentran varios archivos y directorios. El archivo *README* contiene información útil acerca del código y cómo configurar y ejecutar el modelo. Se procede a construir *WRF* con el comando:

```
~$ ./configure
```

Aparece una lista de opciones. Se ingresa la opción 1: Linux i486 i586 i686, gfortrancompilerwithgcc (serial).

```
~$ Enterselction [1-16] : 1
```

Se debe escoger el tipo de anidamiento. La más común es la opción 1.

```
~$ Compile for nesting? (0=no nesting, 1=basic, 2=preset
moves, 3=vortex following) [default 0]: 1
```

Esto crea el archivo *configure.wrf*. Ahora se pasa a la etapa de compilación para lo cual se emplea el comando:

```
~$ ./compile em_real >& compile.log
```

Si la compilación fue exitosa, aparecen los siguientes archivos ejecutables en el directorio *main/*:

main/ndown.exe	(Usado para anidamiento de una vía).
main/nup.exe	(Usado con <i>WRF-Var</i> ).
main/real.exe	(Inicialización <i>WRF</i> para casos de datos reales).
main/wrf.exe	(Modelo de integración <i>WRF</i> ).

Estos ejecutables son enlazados simbólicamente desde el directorio *main/* a los directorios *run/* y *test/em\_real/*, desde donde se ejecuta el código.

### 2.2.2.3 Configuración y compilación del código de *WPS*

Se descarga el código fuente para *WPS*. Este código se almacena en el directorio de trabajo: */home/user/Desktop/wrf1/WRF/*. El archivo descargado se descomprime usando los siguientes comandos:

```
~$ gunzip WPS.3.2.1.TAR.gz
~$ tar -xf WPS.3.2.1.TAR
```

Luego hay que ubicarse dentro del directorio *WPS/*.

```
~$ cd /home/user/Desktop/wrf1/WRF/WPS
```

El siguiente paso crea un archivo de configuración para la computadora en particular en la que se está trabajando.

```
~$ ./configure
```

Aparece una lista de opciones. Se ingresa la opción 14: PC Linux i486 i586 i686, gfortrancompiler, serial (with GRIB2).

Esto crea el archivo *configure.wps*. A continuación se edita el archivo agregando las siguientes opciones:

```
COMPRESSION_LIBS = -L/usr/lib -ljasper -lpng12 -lpng -lz
```

```
COMPRESSION_INC = -I/usr/include
```

```
NCARG_LIBS = -L$(NCARG_ROOT)/lib -lncarg -lncarg_gks -lncarg_c -L/usr/lib/X11 -lX11
```

```
NCARG_LIBS2 = -L/usr/lib -lpng
```

Sigue la etapa de compilación:

```
~$ ./compile >& compile.log
```

Si la compilación fue exitosa, aparecen los siguientes archivos ejecutables:

```
geogrid.exe ->geogrid/src/geogrid.exe (Genera datos estáticos).
```

```
metgrid.exe ->metgrid/src/metgrid.exe (Genera los datos de entrada para WRFV3).
```

```
ungrib.exe ->ungrib/src/ungrid.exe (Descifra los archivos GRIB2).
```



Ya que la herramienta gráfica *NCL* está instalada, los siguientes archivos ejecutables también aparecen:

```
plotgrids.exe -> /util/plotgrids.exe  
plotfmt.exe -> /util/plotfmt.exe
```

Finalmente se modifica el archivo *namelist.wps* para indicar la ubicación de los datos geográficos. Se agrega la siguiente línea a los registros correspondientes a *GEOGRID*:

```
&geogrid  
...  
geog_data_path = '/home/user/Desktop/wrf1/WRF/geog',  
...  
/
```

### 2.2.3 CONFIGURACIÓN DE SERVIDOR *WEB*

En esta sección se presenta los comandos empleados para la instalación y configuración del servidor *Web Apache2*.

Desde la línea de comandos se ejecuta el siguiente código para la instalación de *Apache2*:

```
~$ sudo apt-get install apache2
```

Se habilita *SSL* para el servidor *Web*.

```
~$ sudo a2enmod ssl
```

A continuación se ubica en el directorio de trabajo de *Apache2* donde se está la información del sitio web disponible por defecto:

```
~$ cd /etc/apache2/sites-available
```

Se copia el contenido del archivo *default-ssl* en el archivo *wrfserverssl*.

```
~$ sudo cp default-ssl wrfserverssl
```

Se deshabilita el sitio *Web* por defecto.

```
~$ sudo a2dissite default
```

Se modifican las siguientes dos líneas en el archivo *wrfserverssl* especificando la ubicación del directorio del servidor web: */home/user/Desktop/wrf1/Webserver/*.

```
DocumentRoot /home/user/Desktop/wrf1/Webserver  
<Directory /home/user/Desktop/wrf1/Webserver>
```

Se habilita el nuevo sitio.

```
~$ sudo a2dissite wrfserverssl
```

Finalmente se reinicia el servicio *apache2*.

```
~$ sudo /etc/init.d/apache2 restart
```

## 2.2.4 CONFIGURACIÓN DE SERVIDOR *FTP*

En esta sección se presenta los comandos empleados para la instalación y configuración del servidor *FTP vsftpd*.

Desde la línea de comandos se ejecuta el siguiente código para la instalación del servidor *FTP*:

```
~$ sudo apt-get install vsftpd
```

Se accede al directorio *etc/* y se modifican las siguientes dos líneas en el archivo de configuración *vsftpd.conf* para permitir descargas anónimas y además se indica la ubicación del nuevo directorio *home/* del usuario *FTP*.

```
anonymous_enable=YES  
anon_root=/home/user/Desktop/wrf1/ftpdata
```

Finalmente se reinicia el servicio *vsftpd*.

```
~$ sudo restart vsftpd
```

## 2.3 DISEÑO

En esta sección se describe el diseño de los procesos de entrada y salida de datos del modelo ya instalado. Estos procesos comprenden: la descarga de los datos meteorológicos, el ingreso de estos datos a *WPS*, el ingreso de los datos procesados por *WPS* al módulo de predicción *WRF*, y finalmente la publicación de los datos finales.

### 2.3.1 DESCARGA DE DATOS

En esta sección se presenta la estructura del nombre del bloque de datos meteorológicos a descargar. Luego, de acuerdo a esta estructura se explica el proceso de creación de un directorio y posterior descarga de los datos meteorológicos dentro de ese directorio.

Para la descarga del bloque de datos meteorológicos se escogieron los datos de *NCEP* (<ftp://ftp.ncep.noaa.gov/pub/data/nccf/com/gfs/prod/gfs.YYYYMMDDHH>) cuyo formato es *GFS*. Como se describió en el análisis de requerimientos, estos datos son en tiempo real, de acceso libre, y gracias a su empleo los resultados obtenidos son más cercanos a la realidad.

En este sitio, los datos están divididos en 4 grupos. El nombre del directorio depende de la hora en la cual son puestos a disposición al público como se puede observar en la tabla 2.1:

Tiempo	Nombre de directorio
Medianoche	gfs.YYYYMMDD <b>00</b>
Mañana	gfs.YYYYMMDD <b>06</b>
Mediodía	gfs.YYYYMMDD <b>12</b>
Tarde	gfs.YYYYMMDD <b>18</b>

Tabla 2.1. Nombres de directorios de datos meteorológicos de acuerdo a la hora de disponibilidad.

Por tanto:

- gfs. Corresponde al tipo de dato *Global ForecastSystem*.
- YYYYMMDDHH: La fecha en formato *AñoMesDíaHora*.
- Ejemplo: gfs.2011082600. Corresponde al directorio de 26 de agosto de 2011 salido a la medianoche.

Para la realización de simulaciones se puede escoger cualquiera de los bloques de datos. Para el proyecto en desarrollo se escogieron los datos correspondientes al mediodía para la realización de simulaciones. El proceso de automatización de descarga de estos datos comprende los siguientes pasos:

- Crear un directorio con la fecha actual agregando el sufijo **12**, en concordancia con el nombre del directorio del cual se están obteniendo los datos.
- Acceder al directorio.
- Proceder a la descarga de los datos. Para esto se utilizará el comando *wget*<sup>6</sup> con su opción *-c* ó *--continue*. Esto resulta útil si por alguna razón no prevista la descarga se detiene, entonces se puede retomarla en el punto

<sup>6</sup>wget: El descargador de red no interactivo. Fuente: man wget.

donde se quedó sin el riesgo de perder los datos descargados hasta antes de producirse el fallo. (Ver Anexo D).

### 2.3.2 INGRESO DE DATOS A WPS

En la presente sección se describe el ingreso de datos a *WPS*. Como ya se explicó en el capítulo anterior, *WPS* está formado de tres módulos cuyo rol colectivo es preparar la entrada para el módulo *real* en simulaciones con datos reales. Cada módulo realiza una etapa en la preparación: *geogrid* define dominios e interpola datos estáticos geográficos a las grillas; *ungrib* extrae los campos meteorológicos desde los datos de entrada; y *metgrid* interpola horizontalmente los campos meteorológicos extraídos por *ungrib* a las grillas definidas por *geogrid*. En la figura 2.1 se presentan los programas que forman parte de *WPS* y el correspondiente flujo de datos.

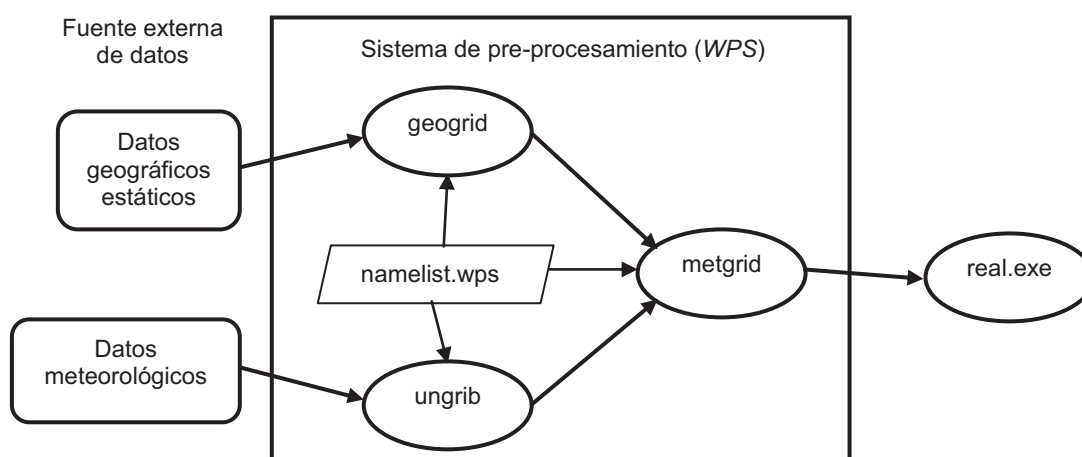


Fig. 2.1. Flujo de datos entre los programas de WPS.<sup>7</sup>

El proceso de automatización en esta etapa abarca los siguientes pasos:

- Ubicarse en el directorio donde reposa el código de *WPS* para empezar con la lectura y descifrado de los datos *GFS* descargados.

<sup>7</sup>Fuente: Chapter 1- Overview, UCAR.

- Se crea un enlace para la *Vtable*<sup>8</sup> relevante y los archivos GRIB de este conjunto de datos. Esta acción se realiza con el programa propio de WPS *link\_gribcsh*.
- Dentro del mismo directorio se copia el contenido del archivo *namelist.wps* en un archivo temporal. En éste último se realizan cambios de fechas y coordenadas de acuerdo a cada dominio propuesto.
- Se define el dominio de simulación ejecutando el módulo *geogrid*.
- Se leen los campos meteorológicos desde los archivos de entrada *GFS* ejecutando el módulo *ungrib*.
- Se interpolan los datos meteorológicos en el modelo ejecutando el módulo *metgrid*.

### 2.3.3 INGRESO DE DATOS AL BLOQUE DE PREDICCIÓN WRF

En esta sección se detallan los pasos a seguir para el ingreso de los datos obtenidos desde WPS en el bloque de predicción WRF. Los casos WRF con datos reales obtienen los datos de entrada desde WPS y son procesados por el módulo *real*. En esta etapa la automatización contempla los pasos descritos a continuación:

- Ubicarse en el directorio de trabajo donde se está el archivo *namelist.input*.
- Se copia el contenido del archivo *namelist.input* a un archivo temporal para realizar los cambios de fechas y coordenadas acordes con los dominios establecidos.
- Ubicarse en el directorio de prueba (*WRFV3/test/*) dentro de WRF para correr el caso real.
- Se crea un enlace simbólico de los archivos *met\_em*<sup>9</sup> creados con el módulo *metgrid* en el sistema de pre-procesamiento.
- Se ejecutan los módulos *real* y *wrf*.

<sup>8</sup>“Variable Tables”. En estas tablas se definen que campos extraer de los archivos de entrada.

<sup>9</sup> La convención es usar “met” para explicar que estos datos son de la salida del programa metgrid de WPS.

### 2.3.4 GRÁFICOS Y PUBLICACIÓN

En esta sección se describe el proceso de visualización y publicación de los datos obtenidos de la simulación. Una vez generado el archivo de salida con la información de la simulación, se procede a ubicarlo en el directorio donde estará disponible al público y luego se procede a su respectiva representación gráfica. Los pasos son los siguientes:

- Se añade la extensión *.nc* al archivo final que ha sido creado para que pueda ser reconocido por la herramienta gráfica *NCL*.
- Se añaden las siguientes letras al nombre del archivo de salida: *\_as\_* para América del Sur, *\_ec\_* para Ecuador y *\_pi\_* para Pichincha. Ejm: ***wrfout\_d01\_as\_YYYY-MM-DD-HH.nc***.
- Se crea un enlace simbólico del archivo final a la ubicación donde será almacenado y mostrado en el servidor *FTP*.
- Ubicarse en el directorio de la página web.
- Se copia el contenido del *script* para gráficos *wrf\_Precip3.ncl* a un archivo temporal para realizar cambios.
- Se ejecuta el programa *NCL* para generar el archivo de salida en formato *PDF* que contienen los gráficos de la simulación.

Este proceso de automatización se realiza para los tres dominios: América del Sur, Ecuador y Pichincha. La parte correspondiente a la descarga de datos se realiza al inicio del proceso de automatización y una sola vez por día.

### 2.4 SCRIPTS DE AUTOMATIZACIÓN

Todo el proceso de automatización descrito en palabras en la sección anterior, será implementado en varios *scripts*: el *script* general y los *scripts* para los tres dominios. El *script* general hace una llamada a cada uno de los *scripts* de dominio. En cada *script* de dominio se realizan las siguientes tareas: modificación de archivos de configuración, ejecución de los módulos de *WPS* y *WRF*, graficación y publicación.

### 2.4.1 *SCRIPT* GENERAL

Este *script* denominado *General.sh* se encuentra en el directorio: */home/user/scripts/* y hace el llamado a los *scripts* que definen los procesos de entrada y salida para cada dominio establecido. Empieza por América del Sur, sigue con Ecuador y termina en Pichincha (ver Anexo C).

### 2.4.2 *SCRIPTS* PARA AMÉRICA DEL SUR

Dentro del directorio: */home/user/scripts/scripts-americadelsur/* se encuentra la siguiente lista de *scripts*:

1\_descarga-datos.sh: Correspondiente con la sección de descarga de datos obtenida del diseño de los procesos de entrada y salida de datos (ver Anexo D).

2\_cambiar\_namelistwps\_as.sh, 3\_correr\_programaswps\_as.sh: Corresponden a la sección de ingreso de datos a *WPS* obtenido del diseño de los procesos de entrada y salida de datos (ver Anexo E y F).

4\_cambia\_namelistinput\_as.sh, 5\_correr\_programaswrf\_as.sh: Corresponden a la sección de ingreso de datos al bloque de predicción *WRF* obtenido del diseño de los procesos de entrada y salida de datos (ver Anexo G y H).

6\_graficar\_as.sh: Correspondiente a la sección de gráficos y publicación obtenida del diseño de procesos de entrada y salida de datos (ver Anexo I).

### 2.4.3 *SCRIPTS* PARA ECUADOR

Los siguientes *scripts* se encargan solamente del ingreso de datos en los módulos *WPS* y *WRF* para el dominio de Ecuador. Dentro del directorio: */home/user/scripts/scripts-ecuador/* se encuentra la siguiente lista de *scripts*:



1\_cambiar\_namelistwps\_ec.sh, 2\_correr\_programaswps\_ec.sh: Corresponden a la sección de ingreso de datos a *WPS* obtenida del diseño de los procesos de entrada y salida de datos (ver Anexo J y K).

3\_cambia\_namelistinput\_ec.sh, 4\_correr\_programaswrf\_ec.sh: Corresponden a la sección de ingreso de datos al bloque de predicción *WRF* obtenido del diseño de los procesos de entrada y salida de datos (ver Anexo L y M).

5\_graficar\_ec.sh: Correspondiente a la sección de gráficos y publicación obtenida del diseño de procesos de entrada y salida de datos (ver Anexo N).

#### **2.4.4 SCRIPTS PARA PICHINCHA**

Los siguientes *scripts* se encargan solamente del ingreso de datos en los módulos *WPS* y *WRF* para el dominio Pichincha. Dentro del directorio: */home/user/scripts/scripts-pichincha/* se encuentra la siguiente lista de *scripts*:

1\_cambiar\_namelistwps\_pi.sh, 2\_correr\_programaswps\_pi.sh: Corresponden a la sección de ingreso de datos a *WPS* obtenida del diseño de los procesos de entrada y salida de datos (ver Anexo O y P).

3\_cambia\_namelistinput\_pi.sh, 4\_correr\_programaswrf\_pi.sh: Corresponden a la sección de ingreso de datos al bloque de predicción *WRF* obtenida del diseño de los procesos de entrada y salida (ver Anexo Q y R).

5\_graficar\_pi.sh: Correspondiente a la sección de gráficos y publicación obtenida del diseño de procesos de entrada y salida de datos (ver Anexo S).

## CAPÍTULO 3

### VALIDACIÓN Y PUBLICACIÓN

En este capítulo se describen las pruebas realizadas con los elementos funcionales que componen el proceso manual y automatizado del modelo *WRF*; adicionalmente se detalla el desarrollo de dichas pruebas y los resultados obtenidos con sus respectivos análisis.

Las pruebas realizadas con los elementos funcionales que componen el proceso manual muestran todas las tareas de ejecución del modelo y su grado de complejidad, esto con el propósito de hacerle saber al lector lo extenso que resulta ejecutar este modelo. Los elementos funcionales que componen el proceso manual del modelo *WRF* abarcan dos escenarios en su fase de prueba: el primero con datos idealizados y el segundo con datos reales.

En el proceso manual con datos idealizados, el usuario debe encargarse de realizar las siguientes tareas: compilar el código fuente *WRF*, escoger uno de los casos ideales de los que se incluyen en el paquete de instalación del modelo y compilarlo, y utilizar la herramienta *NCL* para graficar los resultados obtenidos.

En el proceso manual con datos reales, el usuario debe encargarse de realizar las siguientes tareas: compilar el código fuente *WRF* y *WPS*, especificar la ubicación de datos geográficos y meteorológicos iniciales, ejecutar los tres programas que son parte del sistema de pre-procesamiento, ejecutar los programas de predicción, y utilizar la herramienta *NCL* para graficar los resultados obtenidos.

Por tanto, la obtención de los datos de salida de las simulaciones en el proceso manual depende de dos factores que son: la correcta instalación de los paquetes básicos de software, y la correcta instalación, compilación y funcionamiento del código fuente *WRF* y *WPS*. Además, en cada etapa de compilación se debe

comprobar la creación de los archivos intermedios que son necesarios para el funcionamiento del modelo.

En el proceso automatizado se debe verificar el correcto funcionamiento de los diferentes *scripts* de automatización que se encargan de todas las tareas que realiza el modelo *WRF* empleando datos reales, facilitando así la experiencia del usuario. Se debe comprobar la creación de los archivos intermedios, la ejecución de los programas del sistema de pre-procesamiento y la ejecución de los programas de predicción. Finalmente se debe comparar los resultados de las simulaciones con imágenes satelitales.

Para la realización de estas pruebas se ha definido un protocolo sistemático que permite identificar rápidamente los elementos claves de las mismas. A continuación se puede apreciar la información que es presentada en cada prueba:

**Nombre y descripción de la Prueba:** Este ítem se refiere al nombre que se debe asignar a la prueba e indica una breve descripción de las tareas que se realizan en la prueba.

**Objetivo de la Prueba:** Este ítem detalla los objetivos que se persiguen con la realización de la prueba.

**Procedimiento para realizar la Prueba:** En este ítem se enumeran y explican los pasos a seguir para la realización de la prueba.

**Resultados Esperados:** En este ítem se describen los resultados que se esperan de la prueba.

**Resultados Obtenidos:** En este ítem se describen los resultados que se obtienen de la prueba.

**Análisis de Resultados:** En este ítem se analizan los resultados obtenidos de la prueba.

### 3.1 CASO IDEAL

#### Nombre y descripción de la Prueba

**Prueba del proceso manual con datos idealizados.-** En esta prueba se compilará manualmente un caso con datos idealizados y se realizará la respectiva interpretación gráfica con la herramienta *NCL*. Adicionalmente se debe verificar la creación de un archivo intermedio, mismo que es necesario para la obtención del archivo de salida de la simulación.

#### Objetivo de la Prueba

Comprobar el funcionamiento del proceso manual del modelo *WRF* utilizando datos ideales.

#### Procedimiento para realizar la Prueba

1. La configuración utilizada en este modelo es la misma que se explicó en el Capítulo anterior en el apartado 2.2.2.2.

2. De acuerdo a la estructura de directorios presentada en el apartado 2.1 del Capítulo anterior, se ha iniciado desde el directorio */home/user/Desktop/wrf1/WRF/*. Para la compilación se ha escogido el caso ideal correspondiente a *ondas baroclínicas*:

```
~$ ./compile em_b_wave >& compile.log.
```

3. Posteriormente se procede a ubicarse en el directorio donde se ejecuta el código:

```
~$ cd run/
```

4. A continuación se debe ejecutar el programa *ideal.exe*.

```
~$ ./ideal.exe
```

5. Luego se debe ejecutar el programa *wrf.exe*.

```
~$ ./wrf.exe
```

6. Se debe copiar el archivo generado *wrfout\_d01\_0001-01-01\_00:00:00* a otro archivo con el mismo nombre pero añadiendo al final la extensión *.nc*. Esto permite que el archivo sea reconocido por la herramienta gráfica *NCL*.

7. Se modifica el *script wrf\_BWave.ncl* obtenido de la página del tutorial en línea *WRF*:

- [http://www.mmm.ucar.edu/wrf/OnLineTutorial/Graphics/NCL/Examples/wrf\\_BWave.htm](http://www.mmm.ucar.edu/wrf/OnLineTutorial/Graphics/NCL/Examples/wrf_BWave.htm).

En primer lugar se debe indicar la ubicación del archivo de salida de la simulación. Y en segundo lugar, se escoge el formato de salida de los gráficos. En nuestro caso se generará un archivo formato *PDF*.

Con un editor de texto se abre el archivo *wrf\_BWave.ncl* y la línea:

```
a = addfile("../IDEAL/wrfout_bwave.nc", "r")
```

Se reemplaza por:

```
a = addfile("/home/user/Desktop/wrf1/WRF/WRFV3/run/wrfout_d01_0001-01-01_00:00:00.nc", "r")
```

Esto para indicar a la herramienta gráfica *NCL* la nueva ubicación del archivo de salida de la simulación.

Se comenta el campo: *type = "x11"* anteponiendo el símbolo de punto y coma:

```
; type = "x11".
```

Y se remueve el comentario correspondiente a *PDF*, para que los gráficos generados se presenten en un archivo con formato *PDF*:

```
type = "pdf"
```

11. A través de la línea de comandos se debe ejecutar el *script wrf\_BWave.ncl* que permite obtener gráficamente los resultados de la simulación.

```
~$ ncl wrf_BWave.ncl
```

En la figura 3.1 se muestra el archivo de configuración *wrf\_BWave.ncl* y las líneas modificadas se encuentran resaltadas en color amarillo.

```

; Example script to produce plots for a WRF real-data run,
; with the ARW coordinate dynamics option.
; In this example we first get the entire field over time, which will
; make it easier to calculate tendencies

load "$NCARG_ROOT/lib/ncarg/nclscripts/csm/gsn_code.ncl"
load "$NCARG_ROOT/lib/ncarg/nclscripts/wrf/WRFUserARW.ncl"

begin
;
; The WRF ARW input file.
; This needs to have a ".nc" appended, so just do it.
a = addfile("/home/pancho/Desktop/wrf1/WRF/WRFV3/run/wrfout_d01_001-01-
01_00:00:00.nc", "r")

; We generate plots, but what kind do we prefer?
; type = "x11"
type = "pdf"
; type = "ps"
; type = "ncgm"
wks = gsn_open_wks(type, "Ideal")

; Set some basic resources
res = True
res@MainTitle = "REAL-TIME WRF"

pltres = True
mpres = True
mpres@mpGeophysicalLineColor = "Black"
mpres@mpNationalLineColor = "Black"
mpres@mpUSStateLineColor = "Black"
mpres@mpGridLineColor = "Black"
mpres@mpLimbLineColor = "Black"
mpres@mpPerimLineColor = "Black"

;;;;;;;;;;;;;
; First get the variables we will need

slp = wrf_user_getvar(a, "slp", -1) ; slp
wrf_smooth_2d( slp, 3 ) ; smooth slp

; Get non-convective, convective
; Calculate total precipitation
rain_exp = wrf_user_getvar(a, "RAINNC", -1)
rain_con = wrf_user_getvar(a, "RAINNC", -1)
rain_tot = rain_exp + rain_con
rain_tot@description = "Total Precipitation"

;;;;;;;;;;;;;

; What times and how many time steps are in the data set?
times = wrf_user_list_times(a) ; get times in the file
ntimes = dimsizes(times) ; number of times in the file

;;;;;;;;;;;;;

do it = 1, ntimes-1 ; TIME LOOP - start at hour 3 as we interested
in 3hourly tendencies

```

```

print("Working on time: " + times(it) )
res@TimeLabel = times(it) ; Set Valid time to use on plots

rain_exp_tend = rain_exp(it,::) - rain_exp(it-1,::)
rain_con_tend = rain_con(it,::) - rain_con(it-1,::)
rain_tot_tend = rain_tot(it,::) - rain_tot(it-1,::)
  rain_exp_tend@description = "Explicit Precipitation Tendency"
  rain_con_tend@description = "Param Precipitation Tendency"
  rain_tot_tend@description = "Precipitation Tendency"

; Plotting options for Precipitation
opts_r = res
opts_r@UnitLabel          = "mm"
opts_r@cnLevelSelectionMode = "ExplicitLevels"
opts_r@cnLevels           = (/ .1, .2, .4, .8, 1.6, 3.2, 6.4, \
  12.8, 25.6, 51.2, 102.4/)
opts_r@cnFillColors       = (/ "White", "White", "DarkOliveGreen1", \
  "DarkOliveGreen3", "Chartreuse", \
  "Chartreuse3", "Green", "ForestGreen", \
  "Yellow", "Orange", "Red", "Violet"/)

opts_r@cnInfoLabelOn     = False
opts_r@cnConstFLabelOn  = False
opts_r@cnFillOn          = True

; Total Precipitation (color fill)
contour_tot = wrf_contour(a,wks, rain_tot(it,::), opts_r)

; Precipitation Tendencies
opts_r@SubFieldTitle = "from " + times(it-1) + " to " + times(it)

contour_tend = wrf_contour(a,wks, rain_tot_tend,opts_r) ; total (color)
contour_res = wrf_contour(a,wks,rain_exp_tend,opts_r) ; exp (color)
opts_r@cnFillOn = False
opts_r@cnLineColor = "Red4"
contour_prm = wrf_contour(a,wks,rain_con_tend,opts_r) ; con (red lines)
delete(opts_r)

; MAKE PLOTS

; Total Precipitation
plot = wrf_map_overlays(a,wks,contour_tot,pltres,mpres)

; Non-Convective and Convective Precipitation Tendencies
plot = wrf_map_overlays(a,wks,(/contour_res,contour_prm/),pltres,mpres)

end do ; END OF TIME LOOP

;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
end

```

Fig. 3.1. Archivo de configuración *wrf\_BWave.ncl*.

## Resultados Esperados

1. Si la compilación fue exitosa, deben aparecer los siguientes archivos ejecutables:

main/ideal.exe           (Inicialización *WRF* para casos con datos ideales).  
main/wrf.exe            (Modelo de integración *WRF*).

2. Luego de la ejecución del programa *ideal.exe* se debe crear el archivo *wrfinput\_d01*.

3. Después de la ejecución correcta del programa *wrf.exe*, debe aparecer el archivo *wrfout\_d01\_0001-01-01\_00:00:00*.

4. Luego de la ejecución correcta del *script wrf\_BWave.ncl*, se debe generar el archivo *plt\_BWave.pdf*. (Ver Anexo T).

## Resultados Obtenidos

Se ha ingresado en el directorio *run/* con la finalidad de verificar la creación de los archivos ejecutables *ideal.exe* y *wrf.exe*, y de los archivos *wrfinput\_d01* y *wrfout\_d01\_0001-01-01\_00:00:00*. En la figura 3.2 se muestra el contenido del directorio *run/* luego de la simulación.

```

pancho@ubuntu: ~/Desktop/WRF/WRFV3/run
File Edit View Search Terminal Help
-rw-r--r-- 1 pancho pancho 186 2011-05-30 21:16 wrf.f90
-rw-r--r-- 1 pancho pancho 1392 2011-05-30 21:16 wrf.o
-rw-r--r-- 1 pancho pancho 84474 2010-07-16 12:04 wrf_SST_ESMF.F
pancho@ubuntu:~/Desktop/WRF/WRFV3/main$ cd ../run/
pancho@ubuntu:~/Desktop/WRF/WRFV3/run$ ls
CAM_ABS_DATA          namelist.output      tr49t67
CAM_AEROPT_DATA       ozone.formatted      tr49t85
co2_trans             ozone_lat.formatted  tr67t85
ETAMPNEW_DATA         ozone_plev.formatted URBPARAM.TBL
ETAMPNEW_DATA_DBL    README.namelist     VEGPARAM.TBL
GENPARAM.TBL         README.tslist        wrf.exe
grib2map.tbl         RRTM_DATA           wrfinput_d01
gribmap.txt          RRTM_DATA_DBL       wrfout_d01_0001-01-01_00:00:00
ideal.exe            RRTMG_LW_DATA
input_jet            RRTMG_LW_DATA_DBL
LANDUSE.TBL         RRTMG_SW_DATA
namelist.input       RRTMG_SW_DATA_DBL
namelist.input.backup SOILPARAM.TBL
pancho@ubuntu:~/Desktop/WRF/WRFV3/run$

```

Fig. 3.2. Archivos ejecutables y archivos de salida generados para el caso ideal.



## **Análisis de Resultados**

En la ejecución manual del modelo con datos ideales no se presentaron inconvenientes, se obtuvieron todos los archivos ejecutables y de salida que debían crearse. El modelo *WRF* presenta varios casos de escenarios idealizados empleados para el análisis de fenómenos meteorológicos. Los casos idealizados no requieren un archivo de condiciones de borde lo que reduce el tiempo de ejecución del modelo.

### **3.2 CASO REAL**

#### **Nombre y descripción de la Prueba**

**Prueba del proceso manual con datos reales.-** En esta prueba se compilará manualmente un caso con datos reales y luego se realizará la respectiva interpretación gráfica con la herramienta *NCL*. Se debe verificar la creación de varios archivos intermedios, la correcta compilación de los programas del sistema de pre-procesamiento y la correcta compilación de los programas de predicción.

#### **Objetivo de la Prueba**

Comprobar el funcionamiento manual del modelo *WRF* utilizando datos reales desde 2013-04-27 hasta 2013-04-29 obtenidos en el sitio web: <http://rda.ucar.edu/dsszone/ds083.2/index.html?g=22012#GRIB2> 2013.04.

#### **Procedimiento para realizar la Prueba**

1. La configuración del modelo es la misma que se explicó en el Capítulo anterior en el apartado 2.2.2.2.
2. De acuerdo a la estructura de directorios presentada en el apartado 2.1 del Capítulo anterior, se debe partir desde el directorio

`/home/user/Desktop/wrf1/WRF/`. Para la compilación se debe ejecutar el siguiente comando:

```
~$ ./compile em_real >& compile.log
```

3. Posteriormente se procede a ubicarse en el directorio `WPS/`.

```
~$ cd /home/user/Desktop/wrf1/WRF/WPS
```

4. Se debe configurar `WPS` con el siguiente comando:

```
~$ ./configure
```

5. Debe aparecer una lista de opciones, donde se debe escoger la opción 14: PC Linux i486 i586 i686, gfortran compiler, serial (with GRIB2). A continuación se debe editar el archivo agregando las siguientes opciones:

```
COMPRESSION_LIBS      =      -L/usr/lib -ljasper -lpng12 -lpng -
lz
COMPRESSION_INC       =      -I/usr/include
NCARG_LIBS            =      -L$(NCARG_ROOT)/lib      -lncarg      -
lncarg_gks -lncarg_c -L/usr/lib/X11 -lX11
NCARG_LIBS2           =      -L/usr/lib -lpng
```

En la figura 3.2 se muestran las opciones de configuración de `WPS`.

```

pancho@ubuntu: ~/Desktop/wrf1/WRF/WPS
File Edit View Search Terminal Help
arch  configure  metgrid      namelist.wps.fire  README
clean  geogrid     namelist.wps  namelist.wps.global ungrib
compile link_grib.csh namelist.wps.all_options namelist.wps.nmm  util
pancho@ubuntu:~/Desktop/wrf1/WRF/WPS$ ./configure
Will use NETCDF in dir: /home/pancho/netcdf/
$JASPERLIB or $JASPERINC not found in environment, configuring to build without grib2 I/O...
-----
Please select from among the following supported platforms.
-----
 1. PC Linux i486 i586 i686, PGI compiler      serial, NO GRIB2
 2. PC Linux i486 i586 i686, PGI compiler      serial
 3. PC Linux i486 i586 i686, PGI compiler      DM parallel, NO GRIB2
 4. PC Linux i486 i586 i686, PGI compiler      DM parallel
 5. PC Linux i486 i586 i686, Intel compiler    serial, NO GRIB2
 6. PC Linux i486 i586 i686, Intel compiler    serial
 7. PC Linux i486 i586 i686, Intel compiler    DM parallel, NO GRIB2
 8. PC Linux i486 i586 i686, Intel compiler    DM parallel
 9. PC Linux i486 i586 i686, g95 compiler,     serial, NO GRIB2
10. PC Linux i486 i586 i686, g95 compiler,     serial
11. PC Linux i486 i586 i686, g95 compiler,     DM PARALLEL, NO GRIB2
12. PC Linux i486 i586 i686, g95 compiler,     DM PARALLEL
13. PC Linux i486 i586 i686, gfortran compiler, serial, NO GRIB2
14. PC Linux i486 i586 i686, gfortran compiler, serial
15. PC Linux i486 i586 i686, gfortran compiler, DM PARALLEL, NO GRIB2
16. PC Linux i486 i586 i686, gfortran compiler, DM PARALLEL

Enter selection [1-16] : 14
-----
Configuration successful. To build the WPS, type: compile
-----
pancho@ubuntu:~/Desktop/wrf1/WRF/WPS$
```

Fig. 3.3. Opciones de ejecución de `WPS`.

6. Se debe compilar *WPS* con el siguiente comando:

```
~$ ./compile >& compile.log
```

7. Se debe editar el archivo *namelist.wps*. Solamente es necesario realizar el siguiente cambio:

```
max_dom = 1
```

8. Se debe ejecutar *geogrid.exe* para crear datos estáticos para este dominio:

```
~$ ./geogrid.exe
```

La figura 3.4 muestra el proceso de ejecución de *geogrid.exe* y el mensaje de ejecución exitosa.

```

pancho@ubuntu: ~/Desktop/wrf1/WRF/WPS
pancho@ubuntu:~/Desktop/wrf1/WRF/WPS$ ./geogrid.exe
Parsed 21 entries in GEOGRID.TBL
Processing domain 1 of 1
Processing XLAT and XLONG
Processing MAPFAC
Processing F and E
Processing ROTANG
Processing LANDUSEF
Calculating landmask from LANDUSEF ( WATER = 16 )
Processing HGT_M
Processing HGT_U
Processing HGT_V
Processing SOILTEMP
Processing SOILCTOP
Processing SCT_DOM
Processing SOILCBOT
Processing SCB_DOM
Processing ALBEDO12M
Processing GREENFRAC
Processing SNOALB
Processing SLOPECAT
Processing SLOPECAT
Processing CON
Processing VAR
Processing OA1
Processing OA2
Processing OA3
Processing OA4
Processing OL1
Processing OL2
Processing OL3
Processing OL4
*****
Successful completion of geogrid.
*****
pancho@ubuntu:~/Desktop/wrf1/WRF/WPS$

```

Fig. 3.4. Proceso de *geogrid.exe* y mensaje de ejecución exitosa.

9. Los datos meteorológicos descargados se almacenarán en el directorio */home/user/Desktop/wrf1/WRF/DATA/*.

10. Se deben enlazar los datos en la correspondiente *Vtable*:

```
~$ ln -sf ungrib/Variable_Tables/Vtable.GFS Vtable
```

11. Se deben enlazar los datos *FNL* haciendo uso del *script link\_grib.csh*:

```
~$ ./link_grib.csh ../DATA/fnl/fnl_201304
```

En la figura 3.5 se presenta el contenido del directorio *WPS* luego de hacer el descifrado y enlazado de los archivos *FNL*.

```

File Edit View Search Terminal Tabs Help
pancho@ubuntu:~/scripts/scripts-america/adsur x pancho@ubuntu:~/Desktop/wrf1/WRF/WPS
arch GRIBFILE.AAZ GRIBFILE.ACI GRIBFILE.ADR GRIBFILE.AFA GRIBFILE.AGJ GRIBFILE.AHS GRIBFILE.AJB GRIBFILE.AKK
clean GRIBFILE.ABA GRIBFILE.ACJ GRIBFILE.ADS GRIBFILE.AFB GRIBFILE.AGK GRIBFILE.AHT GRIBFILE.AJC GRIBFILE.AKL
compile GRIBFILE.ABB GRIBFILE.ACK GRIBFILE.ADT GRIBFILE.AFC GRIBFILE.AGL GRIBFILE.AHU GRIBFILE.AJD GRIBFILE.AKM
compile.log GRIBFILE.ABC GRIBFILE.ACL GRIBFILE.ADU GRIBFILE.AFD GRIBFILE.AGM GRIBFILE.AHV GRIBFILE.AJE GRIBFILE.AKN
configure GRIBFILE.ABD GRIBFILE.ACM GRIBFILE.ADV GRIBFILE.AFE GRIBFILE.AGN GRIBFILE.AHW GRIBFILE.AJF GRIBFILE.AKO
configure.wps GRIBFILE.ABE GRIBFILE.ACN GRIBFILE.ADW GRIBFILE.AFF GRIBFILE.AGO GRIBFILE.AHX GRIBFILE.AJG GRIBFILE.AKP
geo_em.d01.nc GRIBFILE.ABF GRIBFILE.ACO GRIBFILE.ADX GRIBFILE.AFG GRIBFILE.AGP GRIBFILE.AHY GRIBFILE.AJH GRIBFILE.AKQ
geogrid GRIBFILE.ABG GRIBFILE.ACP GRIBFILE.ADY GRIBFILE.AFH GRIBFILE.AGQ GRIBFILE.AHZ GRIBFILE.AJI GRIBFILE.AKR
geogrid.exe GRIBFILE.ABH GRIBFILE.ACQ GRIBFILE.ADZ GRIBFILE.AFI GRIBFILE.AGR GRIBFILE.AIA GRIBFILE.AJJ GRIBFILE.AKS
geogrid.log GRIBFILE.ABI GRIBFILE.ACR GRIBFILE.AEA GRIBFILE.AFJ GRIBFILE.AGS GRIBFILE.AIB GRIBFILE.AJK GRIBFILE.AKT
GRIBFILE.AAA GRIBFILE.ABJ GRIBFILE.ACS GRIBFILE.AEB GRIBFILE.AFK GRIBFILE.AGT GRIBFILE.AIC GRIBFILE.AJL GRIBFILE.AKU
GRIBFILE.AAB GRIBFILE.ABK GRIBFILE.ACT GRIBFILE.AEC GRIBFILE.AFL GRIBFILE.AGU GRIBFILE.AID GRIBFILE.AJM GRIBFILE.AKV
GRIBFILE.AAC GRIBFILE.ABL GRIBFILE.ACU GRIBFILE.AED GRIBFILE.AFM GRIBFILE.AGV GRIBFILE.AIE GRIBFILE.AJN GRIBFILE.AKW
GRIBFILE.AAD GRIBFILE.ABM GRIBFILE.ACV GRIBFILE.AEE GRIBFILE.AFN GRIBFILE.AGW GRIBFILE.AIF GRIBFILE.AJO GRIBFILE.AKX
GRIBFILE.AAE GRIBFILE.ABN GRIBFILE.ACW GRIBFILE.AEF GRIBFILE.AFO GRIBFILE.AGX GRIBFILE.AIG GRIBFILE.AJP GRIBFILE.AKY
GRIBFILE.AAF GRIBFILE.ABO GRIBFILE.ACX GRIBFILE.AEG GRIBFILE.AFP GRIBFILE.AGY GRIBFILE.AIH GRIBFILE.AJQ GRIBFILE.AKZ
GRIBFILE.AAG GRIBFILE.ABP GRIBFILE.ACY GRIBFILE.AEH GRIBFILE.AFQ GRIBFILE.AGZ GRIBFILE.AII GRIBFILE.AJR GRIBFILE.ALA
GRIBFILE.AAH GRIBFILE.ABQ GRIBFILE.ACZ GRIBFILE.AEI GRIBFILE.AFR GRIBFILE.AHA GRIBFILE.AIJ GRIBFILE.AJS link_grib.csh
GRIBFILE.AAI GRIBFILE.ABR GRIBFILE.ADA GRIBFILE.AEJ GRIBFILE.AFS GRIBFILE.AHB GRIBFILE.AIK GRIBFILE.AJT metgrid
GRIBFILE.AAJ GRIBFILE.ABS GRIBFILE.ADB GRIBFILE.AEK GRIBFILE.AFT GRIBFILE.AHC GRIBFILE.AIL GRIBFILE.AJU metgrid.exe
GRIBFILE.AAK GRIBFILE.ABT GRIBFILE.ADC GRIBFILE.AEL GRIBFILE.AFU GRIBFILE.AHD GRIBFILE.AIM GRIBFILE.AJV metgrid.log
GRIBFILE.AAL GRIBFILE.ABU GRIBFILE.ADD GRIBFILE.AEM GRIBFILE.AFV GRIBFILE.AHE GRIBFILE.AIN GRIBFILE.AJW namelist.wps
GRIBFILE.AAM GRIBFILE.ABV GRIBFILE.ADE GRIBFILE.AEN GRIBFILE.AFW GRIBFILE.AHF GRIBFILE.AIO GRIBFILE.AJX namelist.wps.all_options
GRIBFILE.AAN GRIBFILE.ABW GRIBFILE.ADF GRIBFILE.AEO GRIBFILE.AFX GRIBFILE.AHG GRIBFILE.AIP GRIBFILE.AJY namelist.wps.fire
GRIBFILE.AAO GRIBFILE.ABX GRIBFILE.ADG GRIBFILE.AEP GRIBFILE.AFY GRIBFILE.AHH GRIBFILE.AIQ GRIBFILE.AJZ namelist.wps.global
GRIBFILE.AAP GRIBFILE.ABY GRIBFILE.ADH GRIBFILE.AEQ GRIBFILE.AFI GRIBFILE.AHT GRIBFILE.AIR GRIBFILE.AKA namelist.wps.nmm
GRIBFILE.AAQ GRIBFILE.ABZ GRIBFILE.ADI GRIBFILE.AER GRIBFILE.AGA GRIBFILE.AHJ GRIBFILE.AIS GRIBFILE.AKB README
GRIBFILE.AAR GRIBFILE.ACA GRIBFILE.ADJ GRIBFILE.AES GRIBFILE.AGB GRIBFILE.AHK GRIBFILE.AIT GRIBFILE.AKC ungrrib
GRIBFILE.AAS GRIBFILE.ACB GRIBFILE.ADK GRIBFILE.AET GRIBFILE.AGC GRIBFILE.AHL GRIBFILE.AIU GRIBFILE.AKD ungrrib_data.log
GRIBFILE.AAT GRIBFILE.ACC GRIBFILE.ADL GRIBFILE.AEU GRIBFILE.AGD GRIBFILE.AHM GRIBFILE.AIV GRIBFILE.AKE ungrrib.exe
GRIBFILE.AAU GRIBFILE.ACD GRIBFILE.ADM GRIBFILE.AEV GRIBFILE.AGE GRIBFILE.AHN GRIBFILE.AIW GRIBFILE.AKF ungrrib.log
GRIBFILE.AAV GRIBFILE.ACE GRIBFILE.ADN GRIBFILE.AEW GRIBFILE.AGF GRIBFILE.AHO GRIBFILE.AIX GRIBFILE.AKG util
GRIBFILE.AAW GRIBFILE.ACF GRIBFILE.ADO GRIBFILE.AEX GRIBFILE.AGG GRIBFILE.AHP GRIBFILE.AIY GRIBFILE.AKH Vtable
GRIBFILE.AAX GRIBFILE.ACG GRIBFILE.ADP GRIBFILE.AEY GRIBFILE.AGH GRIBFILE.AHQ GRIBFILE.AIZ GRIBFILE.AKI
GRIBFILE.AAY GRIBFILE.ACH GRIBFILE.ADQ GRIBFILE.AEZ GRIBFILE.AGI GRIBFILE.AHR GRIBFILE.AJA GRIBFILE.AKJ
pancho@ubuntu:~/Desktop/wrf1/WRF/WPS$

```

Fig. 3.5. Contenido del directorio *WPS* luego de descifrado y enlazado de archivos *FNL*.

12. Se debe editar el archivo *namelist.wps* cambiando las siguientes variables correspondientes a la fecha de inicio y finalización de la simulación:

```

&share
start_date = '2013-04-27_12:00:00',
end_date = '2013-04-29_12:00:00',
interval_seconds = 21600,
/
&ungrrib
prefix = 'FILE',
/

```

13. Se debe ejecutar el programa *ungrrib.exe* para crear archivos intermedios:

```
~$ ./ungrrib.exe >& ungrrib_data.log
```

14. Se debe ejecutar el programa *metgrid.exe* para interpolar los datos de entrada en el dominio:

```
~$ ./metgrid.exe
```

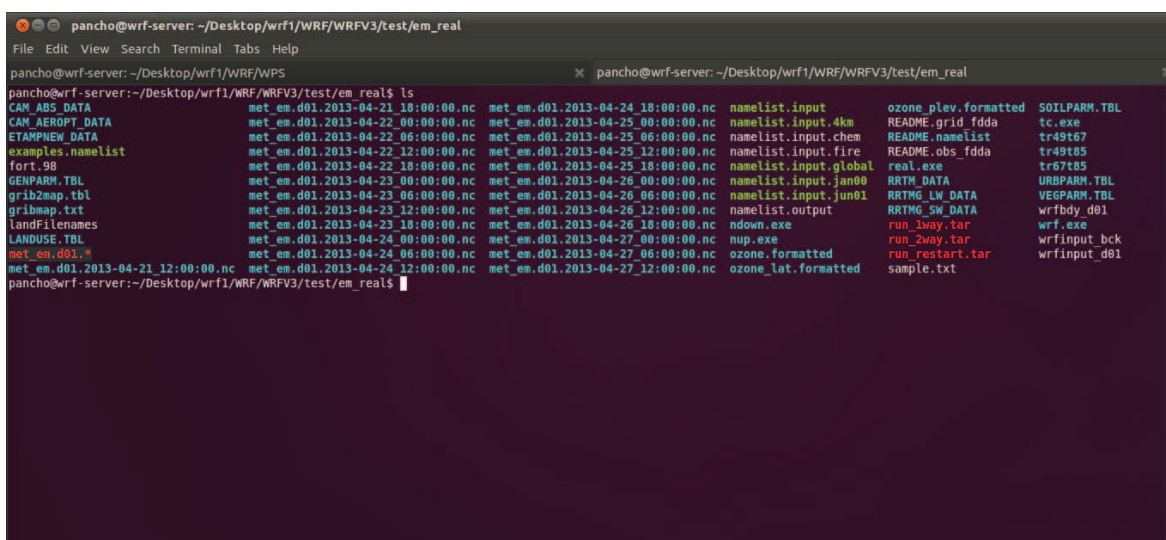
15. Posteriormente se procede a ubicarse en el directorio de trabajo *WRFV3/*:

```
~$ cd ../WRFV3/test/em_real/
```

16. Se deben enlazar a esta ubicación los archivos *met\_em* creados con el programa *metgrid.exe*:

```
~$ ln -sf ../../../../WPS/met_em.d01.2012-0* .
```

En la figura 3.6 se presenta el contenido del directorio *em\_real* luego de hacer el enlazado de los archivos *met\_em*.



```
pancho@wrf-server: ~/Desktop/wrf1/WRF/WRFV3/test/em_real
pancho@wrf-server: ~/Desktop/wrf1/WRF/WRFV3/test/em_real
pancho@wrf-server: ~/Desktop/wrf1/WRF/WRFV3/test/em_real$ ls
CAM_ABS_DATA          met_em.d01.2013-04-21_18:00:00.nc  met_em.d01.2013-04-24_18:00:00.nc  namelist.input          ozone_plav.formatted  SOILPARM.TBL
CAM_AEROPRT_DATA     met_em.d01.2013-04-22_00:00:00.nc  met_em.d01.2013-04-25_00:00:00.nc  namelist.input.4km     README.grid_fdda      tc.exe
CAM_AMPNEW_DATA      met_em.d01.2013-04-22_06:00:00.nc  met_em.d01.2013-04-25_06:00:00.nc  namelist.input.chem    README.namelist      tr49t67
examples.namelist    met_em.d01.2013-04-22_12:00:00.nc  met_em.d01.2013-04-25_12:00:00.nc  namelist.input.fire    README.obs_fdda       tr49t85
fort.98              met_em.d01.2013-04-22_18:00:00.nc  met_em.d01.2013-04-25_18:00:00.nc  namelist.input.global  real.exe              tr67t85
GENPARM.TBL          met_em.d01.2013-04-23_00:00:00.nc  met_em.d01.2013-04-26_00:00:00.nc  namelist.input.jan00   RRTM_DATA             URBPARM.TBL
grib2map.tbl         met_em.d01.2013-04-23_06:00:00.nc  met_em.d01.2013-04-26_06:00:00.nc  namelist.input.jun01  RRTMG_LW_DATA         VEGPARM.TBL
gribmap.txt          met_em.d01.2013-04-23_12:00:00.nc  met_em.d01.2013-04-26_12:00:00.nc  namelist.output       run_1way.tar          wrfbdy_d01
landfilenames        met_em.d01.2013-04-23_18:00:00.nc  met_em.d01.2013-04-26_18:00:00.nc  ndown.exe             run_2way.tar          wrf.exe
LANDUSE.TBL          met_em.d01.2013-04-24_00:00:00.nc  met_em.d01.2013-04-27_00:00:00.nc  nup.exe               run_restart.tar       wrfinput_bck
met_em.d01.*          met_em.d01.2013-04-24_06:00:00.nc  met_em.d01.2013-04-27_06:00:00.nc  ozone.formatted       sample.txt             wrfinput_d01
met_em.d01.2013-04-21_12:00:00.nc  met_em.d01.2013-04-24_12:00:00.nc  met_em.d01.2013-04-27_12:00:00.nc  ozone_lat.formatted
pancho@wrf-server: ~/Desktop/wrf1/WRF/WRFV3/test/em_real$
```

Fig. 3.6. Contenido del directorio *em\_real* luego de enlazado de archivos *met\_em*.

17. Se debe ejecutar el programa *real.exe*:

```
~$ ./real.exe
```

En la figura 3.7 se muestra el mensaje de correcta ejecución de *real.exe*.



```

pancho@wrf-server: ~/Desktop/wrf1/WRF/WRFV3/test/em_real
File Edit View Search Terminal Tabs Help
pancho@wrf-server:~/Desktop/wrf1/WRF/WPS
pancho@wrf-server:~/Desktop/wrf1/WRF/WRFV3/test/em_real
WATER CHANGE = 0
d01 2013-04-27 06:00:00 Total post number of sea ice location changes (water to
land) = 3
d01 2013-04-27 06:00:00 Timing for processing 1 s.
LBC valid between these times 2013-04-27 00:00:00.0000 2013-04-27 06:00:00
d01 2013-04-27 06:00:00 Timing for output 0 s.
d01 2013-04-27 06:00:00 Timing for loop # 24 = 1 s.
-----
Domain 1: Current date being processed: 2013-04-27 12:00:00.0000, which is loo
p # 25 out of 25
configflags%julyr, %julyday, %gmt: 2013 117 12.000000
metgrid input wrf.F first date input = 2013-04-27 12:00:00
metgrid input wrf.F first date nml = 2013-04-21 12:00:00
d01 2013-04-27 12:00:00 Timing for input 0 s.
d01 2013-04-27 12:00:00 flag_soil layers read from met_em file is 1
d01 2013-04-27 12:00:00 Old data, no inland lake information
Assume Noah LSM input
d01 2013-04-27 12:00:00 forcing artificial silty clay loam at 2 points, out
of 16116
LAND CHANGE = 0
WATER CHANGE = 0
d01 2013-04-27 12:00:00 Total post number of sea ice location changes (water to
land) = 3
d01 2013-04-27 12:00:00 Timing for processing 1 s.
LBC valid between these times 2013-04-27 06:00:00.0000 2013-04-27 12:00:00
d01 2013-04-27 12:00:00 Timing for output 0 s.
d01 2013-04-27 12:00:00 Timing for loop # 25 = 1 s.
d01 2013-04-27 12:00:00 real_em: SUCCESS COMPLETE REAL_EM INIT
pancho@wrf-server:~/Desktop/wrf1/WRF/WRFV3/test/em_real$

```

Fig. 3.7. Mensaje de correcta ejecución de *real.exe*.

18. Se debe ejecutar el programa *wrf.exe*:

```
~$ ./wrf.exe
```

En la figura 3.8 se muestra el mensaje de correcta ejecución de *wrf.exe*.

```

pancho@wrf-server: ~/Desktop/wrf1/WRF/WRFV3/test/em_real
File Edit View Search Terminal Tabs Help
pancho@wrf-server:~/Desktop/wrf1/WRF/WPS
pancho@wrf-server:~/Desktop/wrf1/WRF/WRFV3/test/em_real
Timing for main: time 2013-04-22 09:42:54 on domain 1: 7.18500 elapsed seconds.
Timing for main: time 2013-04-22 09:47:57 on domain 1: 12.33800 elapsed seconds.
Timing for main: time 2013-04-22 09:53:00 on domain 1: 7.08600 elapsed seconds.
Timing for main: time 2013-04-22 09:58:03 on domain 1: 7.16600 elapsed seconds.
Timing for main: time 2013-04-22 10:03:06 on domain 1: 7.43700 elapsed seconds.
Timing for main: time 2013-04-22 10:08:09 on domain 1: 7.06400 elapsed seconds.
Timing for main: time 2013-04-22 10:13:12 on domain 1: 7.17300 elapsed seconds.
Timing for main: time 2013-04-22 10:18:15 on domain 1: 12.23600 elapsed seconds.
Timing for main: time 2013-04-22 10:23:18 on domain 1: 7.14800 elapsed seconds.
Timing for main: time 2013-04-22 10:28:21 on domain 1: 7.11600 elapsed seconds.
Timing for main: time 2013-04-22 10:33:24 on domain 1: 7.17000 elapsed seconds.
Timing for main: time 2013-04-22 10:38:27 on domain 1: 7.07100 elapsed seconds.
Timing for main: time 2013-04-22 10:43:30 on domain 1: 7.06300 elapsed seconds.
Timing for main: time 2013-04-22 10:48:33 on domain 1: 12.19100 elapsed seconds.
Timing for main: time 2013-04-22 10:53:36 on domain 1: 7.04500 elapsed seconds.
Timing for main: time 2013-04-22 10:58:39 on domain 1: 7.26600 elapsed seconds.
Timing for main: time 2013-04-22 11:03:42 on domain 1: 7.55600 elapsed seconds.
Timing for main: time 2013-04-22 11:08:45 on domain 1: 7.11900 elapsed seconds.
Timing for main: time 2013-04-22 11:13:48 on domain 1: 7.11100 elapsed seconds.
Timing for main: time 2013-04-22 11:18:51 on domain 1: 12.37700 elapsed seconds.
Timing for main: time 2013-04-22 11:23:54 on domain 1: 7.33700 elapsed seconds.
Timing for main: time 2013-04-22 11:28:57 on domain 1: 7.34600 elapsed seconds.
Timing for main: time 2013-04-22 11:34:00 on domain 1: 7.18200 elapsed seconds.
Timing for main: time 2013-04-22 11:39:03 on domain 1: 7.18500 elapsed seconds.
Timing for main: time 2013-04-22 11:44:06 on domain 1: 7.15100 elapsed seconds.
Timing for main: time 2013-04-22 11:49:09 on domain 1: 12.42800 elapsed seconds.
Timing for main: time 2013-04-22 11:54:12 on domain 1: 7.17100 elapsed seconds.
Timing for main: time 2013-04-22 11:59:15 on domain 1: 7.05900 elapsed seconds.
Timing for main: time 2013-04-22 12:04:18 on domain 1: 7.15500 elapsed seconds.
d01 2013-04-22 12:04:18 wrf: SUCCESS COMPLETE WRF

```

Fig. 3.8. Mensaje de correcta ejecución de *wrf.exe*.

19. Se debe copiar el archivo generado `wrfout_d01_2013-04-27_12:00:00` a otro archivo con el mismo nombre pero añadiendo al final la extensión `.nc`. Esto permite que el archivo sea reconocido por la herramienta gráfica *NCL*.

20. Se modifica el *script wrf\_BWave.ncl* obtenido de la página del tutorial en línea *WRF*:

[http://www.mmm.ucar.edu/wrf/OnLineTutorial/Graphics/NCL/Examples/wrf\\_BWave.htm](http://www.mmm.ucar.edu/wrf/OnLineTutorial/Graphics/NCL/Examples/wrf_BWave.htm). En primer lugar se debe indicar la ubicación del archivo de salida de la simulación. Y en segundo lugar, se escoge el formato de salida de los gráficos. En nuestro caso se generará un archivo formato *PDF*.

Con un editor de texto se abre el archivo *wrf\_BWave.ncl* y la línea:

```
a = addfile("../IDEAL/wrfout_bwave.nc", "r")
```

Se reemplaza por:

```
a = addfile("/home/user/Desktop/wrf1/WRF/WRFV3/run/wrfout_d01_2013-04-27_12:00:00.nc", "r")
```

Esto para indicar a la herramienta gráfica *NCL* la nueva ubicación del archivo de salida de la simulación.

Se comenta el campo: *type = "x11"* anteponiendo el símbolo de punto y coma:

```
; type = "x11".
```

Y se remueve el comentario correspondiente a *PDF*, para que los gráficos generados se presenten en un archivo con formato *PDF*:

```
type = "pdf"
```

21. A través de la línea de comandos se debe ejecutar el *script wrf\_BWave.ncl* que permite obtener gráficamente los resultados de la simulación.

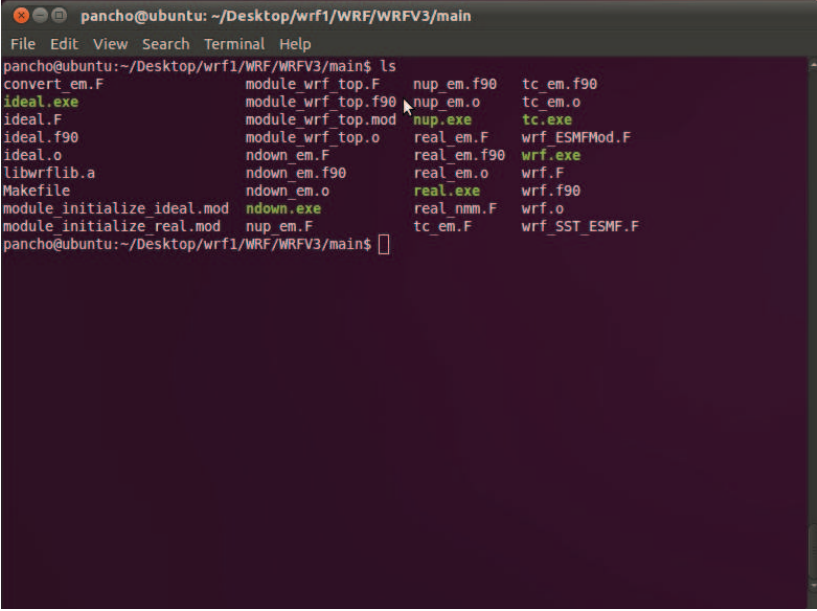
```
~$ ncl wrf_BWave.ncl
```

## Resultados Esperados

1. Si la compilación de *WRF* fue exitosa, deben aparecer los siguientes archivos ejecutables en el directorio *main/*:

main/ndown.exe	(Usado para anidamiento de una vía).
main/nup.exe	(Usado con <i>WRF-Var</i> ).
main/real.exe	(Inicialización <i>WRF</i> para casos de datos reales).
main/wrf.exe	(Modelo de integración <i>WRF</i> ).

En la figura 3.9 se muestra el contenido del directorio *WRF* luego de una compilación exitosa.



```

pancho@ubuntu: ~/Desktop/wrf1/WRF/WRFV3/main
File Edit View Search Terminal Help
pancho@ubuntu:~/Desktop/wrf1/WRF/WRFV3/main$ ls
convert_em.F          module_wrf_top.F      nup_em.f90          tc_em.f90
ideal.exe             module_wrf_top.f90   nup_em.o            tc_em.o
ideal.F               module_wrf_top.mod   nup.exe             tc.exe
ideal.f90             module_wrf_top.o     real_em.F           wrf_ESMFMOD.F
ideal.o               ndown_em.F           real_em.f90         wrf.exe
libwrf.a              ndown_em.f90         real_em.o           wrf.F
Makefile              ndown_em.o           real.exe            wrf.f90
module_initialize_ideal.mod ndown.exe            real_nmm.F          wrf.o
module_initialize_real.mod nup_em.F             tc_em.F             wrf_SST_ESMF.F
pancho@ubuntu:~/Desktop/wrf1/WRF/WRFV3/main$

```

Fig. 3.9. Contenido del directorio *WRF* luego de correcta ejecución de *wrf.exe*.

2. Además se debe crear el archivo *configure.wps*.

3. Si la compilación de *WPS* fue exitosa, deben aparecer los siguientes archivos ejecutables:

*geogrid.exe* -> *geogrid/src/geogrid.exe* (Genera datos estáticos).

*ungrib.exe* -> *ungrib/src/ungrib.exe* (Descifra los archivos *GRIB*).

*metgrid.exe* -> *metgrid/src/metgrid.exe* (Genera los datos de entrada para *WRFV3*).

4. Si la ejecución de *geogrid.exe* no presenta problemas, al final se debe mostrar el siguiente mensaje “*Successful completion of geogrid*”. También se deben crear los archivos *geogrid.log* y *geo\_em.d01.nc*.



5. Si la ejecución de *ungrib.exe* fue exitosa se deben crear los siguientes archivos:

```
FILE:2013-04-27_12
FILE:2013-04-27_18
FILE:2013-04-28_00
FILE:2013-04-28_06
FILE:2013-04-28_12
FILE:2013-04-28_18
FILE:2013-04-29_00
FILE:2013-04-29_06
FILE:2013-04-28_12
```

6. Si la ejecución de *metgrid.exe* no presenta problemas, al final se debe mostrar el siguiente mensaje "*Successful completion of metgrid*". Se debe crear el archivo *metgrid.log* y también los siguientes archivos:

```
met_em.d01.2013-04-27_12:00:00.nc
met_em.d01.2013-04-27_18:00:00.nc
met_em.d01.2013-04-28_00:00:00.nc
met_em.d01.2013-04-28_06:00:00.nc
met_em.d01.2013-04-28_12:00:00.nc
met_em.d01.2013-04-28_18:00:00.nc
met_em.d01.2013-04-29_00:00:00.nc
met_em.d01.2013-04-29_06:00:00.nc
met_em.d01.2013-04-28_12:00:00.nc
```

7. Después de la ejecución de *real.exe*, los siguientes archivos deben crearse:

```
wrfinput_d01
wrfbdy_d01
```

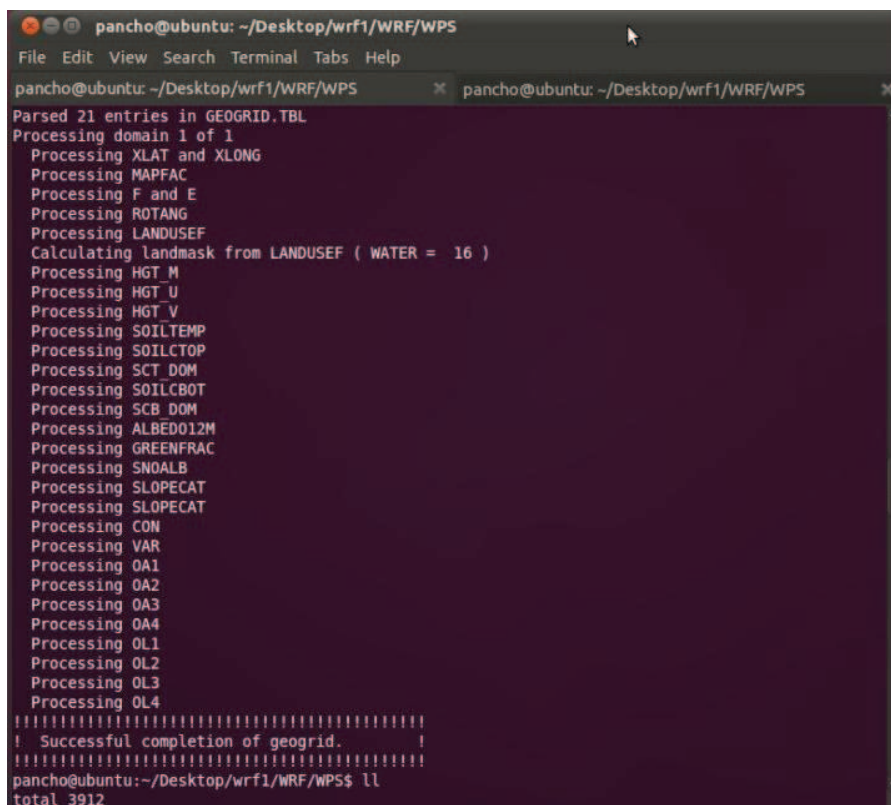
8. Se debe verificar que el siguiente archivo haya sido creado luego de la ejecución de *wrf.exe*:

```
wrfout_d01_2012-08-20_06:00:00
```

9. Luego de la ejecución correcta del *script wrf\_BWave.ncl*, se debe generar el archivo *plt\_BWave.pdf*.

## Resultados Obtenidos

Luego de la compilación de *WPS* se han obtenido los tres archivos ejecutables: *geogrid.exe*, *metgrid.exe* y *unbrib.exe*. A continuación en la figura 3.10 se presenta el mensaje de ejecución exitosa del programa *geogrid.exe*.



```

pancho@ubuntu: ~/Desktop/wrf1/WRF/WPS
File Edit View Search Terminal Tabs Help
pancho@ubuntu: ~/Desktop/wrf1/WRF/WPS x pancho@ubuntu: ~/Desktop/wrf1/WRF/WPS x
Parsed 21 entries in GEOGRID.TBL
Processing domain 1 of 1
  Processing XLAT and XLONG
  Processing MAPFAC
  Processing F and E
  Processing ROTANG
  Processing LANDUSEF
  Calculating landmask from LANDUSEF ( WATER = 16 )
  Processing HGT_M
  Processing HGT_U
  Processing HGT_V
  Processing SOILTEMP
  Processing SOILCTOP
  Processing SCT_DOM
  Processing SOILCBOT
  Processing SCB_DOM
  Processing ALBED012M
  Processing GREENFRAC
  Processing SNOALB
  Processing SLOPECAT
  Processing SLOPECAT
  Processing CON
  Processing VAR
  Processing OA1
  Processing OA2
  Processing OA3
  Processing OA4
  Processing OL1
  Processing OL2
  Processing OL3
  Processing OL4
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
! Successful completion of geogrid. !
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
pancho@ubuntu:~/Desktop/wrf1/WRF/WPS$ ll
total 3912

```

Fig. 3.10. Ejecución exitosa del programa *geogrid.exe*.

En la figura 3.11 se muestra el contenido del directorio *WPS/* para verificar la creación del archivo estático *geo\_em.d01.nc*.

```

pancho@ubuntu: ~/Desktop/wrf1/WRF/WPS
File Edit View Search Terminal Tabs Help
pancho@ubuntu:~/Desktop/wrf1/WRF/WPS x pancho@ubuntu:~/Desktop/wrf1/WRF/WPS x
! Successful completion of geogrid.
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
pancho@ubuntu:~/Desktop/wrf1/WRF/WPS$ ll
total 3912
drwxr-xr-x 14 pancho pancho 20480 2012-08-29 17:35 ./
drwxr-xr-x 6 pancho pancho 4096 2012-07-04 13:25 ../
drwxr-xr-x 2 pancho pancho 4096 2012-07-05 21:02 3Domains/
drwxr-xr-x 2 pancho pancho 4096 2012-08-18 13:08 arch/
-rwxr-xr-x 1 pancho pancho 1714 2012-03-04 16:46 clean*
-rwxr-xr-x 1 pancho pancho 4003 2012-03-31 06:31 compile*
-rw-r--r-- 1 pancho pancho 95321 2012-06-07 16:04 compile.log
-rwxr-xr-x 1 pancho pancho 4599 2012-01-13 13:15 configure*
-rw-r--r-- 1 pancho pancho 2780 2012-06-07 16:03 configure.wps
drwxr-xr-x 2 pancho pancho 4096 2012-06-24 13:16 datos-prueba-real-ok/
drwxr-xr-x 2 pancho pancho 4096 2012-08-03 20:12 EC/
drwxr-xr-x 2 pancho pancho 4096 2012-06-24 12:12 Ecuador/
-rw-r--r-- 1 pancho pancho 2171652 2012-08-20 17:35 geo_em.d01.nc

```

Fig. 3.11. Archivo `geo_em.d01.nc` creado con información del dominio.

Posteriormente, luego de la ejecución de `unrib.exe` y `metgrid.exe`, en el directorio de trabajo `test/em_real/` se ha obtenido el archivo final de simulación: `wrfout_d01_2013-04-27_12:00:00`.

### Análisis de Resultados

Los pasos esenciales durante la realización de esta prueba corresponden a la ejecución de los programas `geogrid.exe`, `ungrib.exe` y `metgrid.exe`, en este estricto orden. Se debe prestar atención en estos programas ya que están interconectados. La ejecución incorrecta de `geogrid.exe` no permitirá pasar a la ejecución de `ungrib.exe`, la ejecución incorrecta de `ungrib.exe` no generará los archivos intermedios necesarios para el funcionamiento y ejecución de `metgrid.exe`, y la incorrecta ejecución de `metgrid.exe` no generará los archivos intermedios necesarios para el funcionamiento del programa de predicción.

Se debe verificar que las fechas de inicio y finalización de la simulación sean las mismas en los dos archivos de configuración: `namelist.wps` y `namelist.input`, pues esto generará errores al momento de ejecutar el programa `metgrid.exe` y no se podrán generar los archivos intermedios necesarios para el siguiente bloque.

Finalmente, el archivo generado: `wrfout_d01_2013-05-27_12:00:00`, se debe copiar a otro archivo con el mismo nombre pero añadiendo al final la

extensión *.nc*. Esto permite que el archivo sea reconocido por la herramienta gráfica *NCL*.

### 3.3 MODELO AUTOMATIZADO

#### Nombre y descripción de la Prueba

**Prueba del Modelo Automatizado.-** En esta prueba se compilará de forma automática un caso con datos reales. Se verificará el correcto funcionamiento de los *scripts* de automatización. Además se comprobará la creación de varios archivos intermedios, la correcta compilación de los programas del sistema de pre-procesamiento, la correcta compilación de los programas de predicción, y la presentación de los archivos finales en el sitio web.

#### Objetivo de la Prueba

Verificar el correcto funcionamiento del modelo automatizado realizando una simulación con datos reales.

#### Procedimiento para realizar la Prueba

1. De acuerdo a la estructura de directorios presentada en el apartado 2.1 del Capítulo anterior, se debe ingresar en el directorio de trabajo */home/user/scripts/* donde se encuentran todos los *scripts* de automatización, para ejecutar el archivo *General.sh*.

```
~$ cd /home/user/scripts/
```

```
~$ ./General.sh
```

2. Se debe esperar hasta que se realice la descarga de datos y el procesamiento *WPS* y *WRF*. El período de simulación va desde 2013-05-03\_12 hasta 2013-05-05\_12.

## Resultados Esperados

Automáticamente debe realizarse la descarga de datos meteorológicos, el procesamiento *WPS*, la modificación de los archivos de configuración, el procesamiento *WRF*, la interpretación gráfica de los datos de salida con la herramienta *NCL* y la publicación de los resultados en la página web.

## Resultados Obtenidos:

Durante el proceso de automatización correspondiente al programa *geogrid.exe* no se presentan problemas de ejecución. En la figura 3.12 se muestra el mensaje “Successful completion of geogrid” que aparece al final del proceso.

```

pancho@wrf-server: ~/scripts02
File Edit View Search Terminal Help
pancho@wrf-server:~$ cd scripts02/
pancho@wrf-server:~/scripts02$ ./General.sh
Parsed 21 entries in GEOGRID.TBL
Processing domain 1 of 1
Processing XLAT and XLONG
Processing MAPPAC
Processing F and E
Processing ROTANG
Processing LANDUSEF
Calculating Landmask from LANDUSEF ( WATER = 16 )
Processing HGT M
Processing HGT U
Processing HGT V
Processing SOILTEMP
Processing SOILCTOP
Processing SCT DOM
Processing SOILCBOT
Processing SCB DOM
Processing ALBEDO12M
Processing GREENFRAC
Processing SNOWLB
Processing SLOPECAT
Processing SLOPECAT
Processing CON
Processing VAR
Processing OA1
Processing OA2
Processing OA3
Processing OA4
Processing OL1
Processing OL2
Processing OL3
Processing OL4
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
! Successful completion of geogrid. !
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!

```

Fig. 3.12. Automatización. Programa *geogrid.exe*.

Durante el proceso de automatización correspondiente al programa *metgrid.exe* no se presentan problemas. La figura 3.13 muestra el mensaje de procesamiento de dominio con los datos *GFS* y el mensaje final de correcta ejecución de *metgrid.exe*: “Successful completion of metgrid”.

```

Applications Places System
pancho@wrf-server: ~/scripts02
File Edit View Search Terminal Help
Processing domain 1 of 1
Processing 2011-05-03_12
GFS2
Processing 2011-05-03_18
GFS2
Processing 2011-05-04_00
GFS2
Processing 2011-05-04_06
GFS2
Processing 2011-05-04_12
GFS2
Processing 2011-05-04_18
GFS2
Processing 2011-05-05_00
GFS2
Processing 2011-05-05_06
GFS2
Processing 2011-05-05_12
GFS2
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
! Successful completion of metgrid. !
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!

```

Fig. 3.13. Automatización. Programa *metgrid.exe*.

Durante el proceso de automatización correspondiente al programa *real.exe* y *wrf.exe* se procesan los datos obtenidos a la salida de *metgrid.exe* de acuerdo a las fechas de inicio y finalización de simulación establecidas en los archivos *namelist.input* e *namelist.wps*. En la figura 3.14 se muestran los mensajes a la salida del procesamiento de *real.exe*. Estos mensajes incluyen información de: versión del modelo *WRF*, opciones dinámicas del modelo, datos de inicialización del suelo, y el procesamiento del bloque de datos acorde al intervalo de tiempo de simulación definido en los archivos *namelist.input* y *namelist.wps*.

```

File Edit View Search Terminal Tabs Help
pancho@wrf-server: ~/Desktop/wrf1/WRF/WPS
Timing for main: time 2013-05-05 09:42:54 on domain 1: 7.10500 elapsed seconds.
Timing for main: time 2013-05-05 09:47:57 on domain 1: 12.33800 elapsed seconds.
Timing for main: time 2013-05-05 09:53:00 on domain 1: 7.08600 elapsed seconds.
Timing for main: time 2013-05-05 09:58:03 on domain 1: 7.16600 elapsed seconds.
Timing for main: time 2013-05-05 10:03:06 on domain 1: 7.43700 elapsed seconds.
Timing for main: time 2013-05-05 10:08:09 on domain 1: 7.06400 elapsed seconds.
Timing for main: time 2013-05-05 10:13:12 on domain 1: 7.17300 elapsed seconds.
Timing for main: time 2013-05-05 10:18:15 on domain 1: 12.23600 elapsed seconds.
Timing for main: time 2013-05-05 10:23:18 on domain 1: 7.14800 elapsed seconds.
Timing for main: time 2013-05-05 10:28:21 on domain 1: 7.11600 elapsed seconds.
Timing for main: time 2013-05-05 10:33:24 on domain 1: 7.17000 elapsed seconds.
Timing for main: time 2013-05-05 10:38:27 on domain 1: 7.07100 elapsed seconds.
Timing for main: time 2013-05-05 10:43:30 on domain 1: 7.06300 elapsed seconds.
Timing for main: time 2013-05-05 10:48:33 on domain 1: 12.19100 elapsed seconds.
Timing for main: time 2013-05-05 10:53:36 on domain 1: 7.04500 elapsed seconds.
Timing for main: time 2013-05-05 10:58:39 on domain 1: 7.26600 elapsed seconds.
Timing for main: time 2013-05-05 11:03:42 on domain 1: 7.55600 elapsed seconds.
Timing for main: time 2013-05-05 11:08:45 on domain 1: 7.11900 elapsed seconds.
Timing for main: time 2013-05-05 11:13:48 on domain 1: 7.11100 elapsed seconds.
Timing for main: time 2013-05-05 11:18:51 on domain 1: 12.37700 elapsed seconds.
Timing for main: time 2013-05-05 11:23:54 on domain 1: 7.33700 elapsed seconds.
Timing for main: time 2013-05-05 11:28:57 on domain 1: 7.34600 elapsed seconds.
Timing for main: time 2013-05-05 11:34:00 on domain 1: 7.18200 elapsed seconds.
Timing for main: time 2013-05-05 11:39:03 on domain 1: 7.10500 elapsed seconds.
Timing for main: time 2013-05-05 11:44:06 on domain 1: 7.15100 elapsed seconds.
Timing for main: time 2013-05-05 11:49:09 on domain 1: 12.42800 elapsed seconds.
Timing for main: time 2013-05-05 11:54:12 on domain 1: 7.17100 elapsed seconds.
Timing for main: time 2013-05-05 11:59:15 on domain 1: 7.05900 elapsed seconds.
Timing for main: time 2013-05-05 12:04:18 on domain 1: 7.15500 elapsed seconds.
d01 2013-05-05 12:04:18 wrf: SUCCESS COMPLETE WRF
pancho@wrf-server:~/Desktop/wrf1/WRF/WRFV3/test/em_real$

```

Fig. 3.14. Automatización. Programas *real.exe* y *wrf.exe*.



En el proceso de automatización correspondiente a la herramienta gráfica *NCL* se toma el bloque de datos finales después de la ejecución de los programas *real.exe* y *wrf.exe*. En la figura 3.15 se presenta la salida de procesamiento *NCL* en la línea de comando, donde se muestran las fechas de procesamiento, finalmente se obtiene un documento en formato *PDF* a la salida de *NCL*.

```
Copyright (C) 1995-2011 - All Rights Reserved
University Corporation for Atmospheric Research
NCAR Command Language Version 6.0.0
The use of this software is governed by a License Agreement.
See http://www.ncl.ucar.edu/ for more details.

Variable: times
Type: string
Total Size: 36 bytes
Number of Dimensions: 1
Dimensions and sizes: [9]
Coordinates:
Number of Attributes: 2
  description : times in file
  FillValue : missing
(0) 2011-05-04 12:00:00
(1) 2011-05-04 15:01:48
(2) 2011-05-04 18:03:36
(3) 2011-05-04 21:05:24
(4) 2011-05-05 00:07:12
(5) 2011-05-05 03:09:00
(6) 2011-05-05 06:10:48
(7) 2011-05-05 09:12:36
(8) 2011-05-05 12:14:24
(9) Working on time: 2011-05-04 15:01:48
(9) Working on time: 2011-05-04 18:03:36
(9) Working on time: 2011-05-04 21:05:24
(9) Working on time: 2011-05-05 00:07:12
(9) Working on time: 2011-05-05 03:09:00
(9) Working on time: 2011-05-05 06:10:48
(9) Working on time: 2011-05-05 09:12:36
(9) Working on time: 2011-05-05 12:14:24
```

Fig. 3.15. Automatización. Herramienta gráfica *NCL*.

Luego de la utilización de la herramienta gráfica *NCL* se obtienen tres documentos en formato *PDF*, correspondientes a los tres dominios de simulación propuestos en este proyecto: América del Sur, Ecuador y Pichincha. La figura 3.16 muestra el sitio web donde se presentan los archivos mencionados y el Anexo V presenta el archivo *index.html* del sitio web.

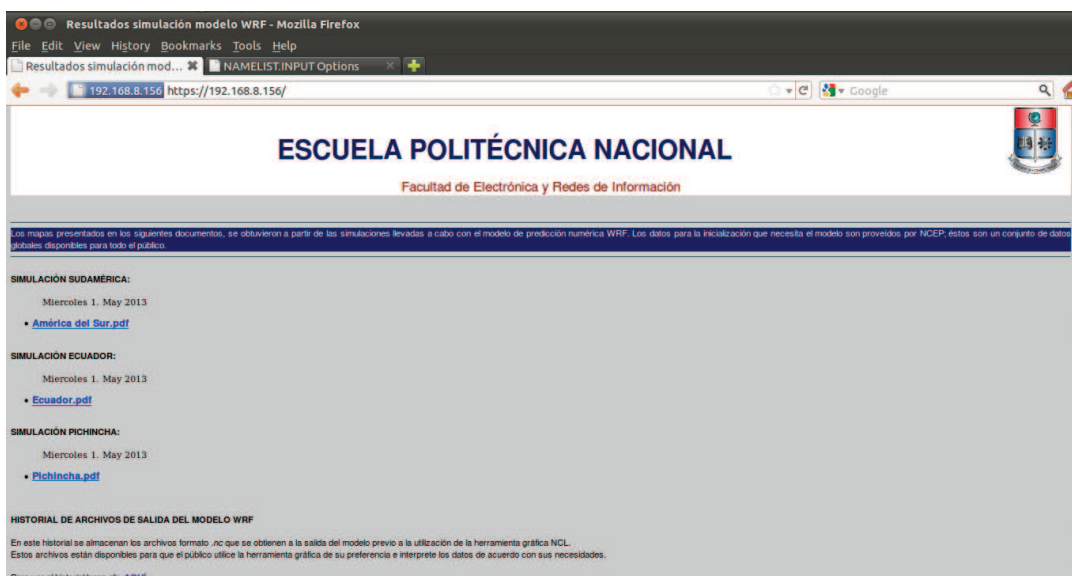


Fig. 3.16. Automatización. Página Web.

Finalmente, el bloque final de datos obtenido a la salida de los programas *wrf.exe* y *real.exe* se ubica automáticamente en un sitio *ftp* que contiene el historial de todos los archivos de simulación. El objetivo de mantener este historial es permitir a todos los usuarios tener alcance a estos archivos para que los procesen de acuerdo a sus propios requerimientos. La figura 3.17 presenta el contenido del sitio *ftp*.

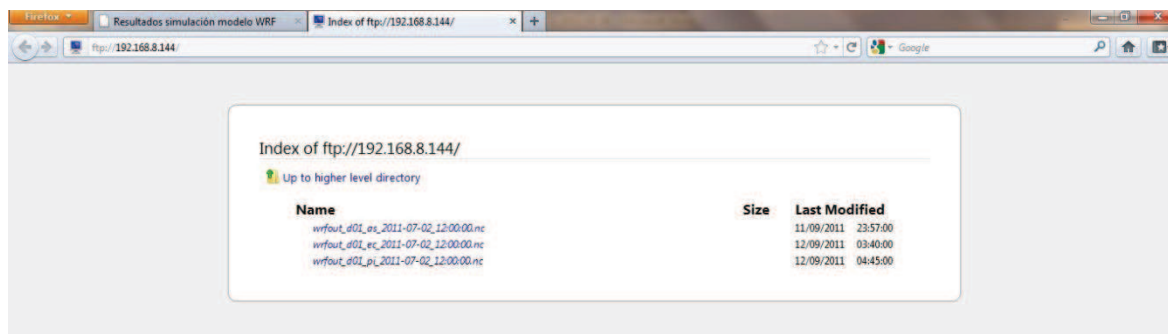


Fig. 3.17. Automatización. Sitio ftp con el historial de archivos.

La información gráfica de las simulaciones se puede verificar en el anexo U.

## Análisis de Resultados

Los resultados obtenidos estuvieron acorde al diseño de los procesos de entrada y salida del modelo. Con esto se comprueba que el usuario no tendrá que preocuparse de:

- Ingresar a la página de descarga de datos meteorológicos,
- Modificar archivos de configuración,
- Ejecutar cada uno de los programas del modelo,
- Graficar los datos de simulación obtenidos y publicarlos en el sitio web.

De todo esto se encargarán los diferentes *scripts* que en el Capítulo anterior se especificaron. Finalmente, en las figuras 3.18 y 3.19 se presentan imágenes satelitales y en la figura 3.20 el resultado de la precipitación total correspondiente a las 15 horas del 2013-05-04 para América del Sur.

La figura 3.18 corresponde a la imagen satelital obtenida el 2013-05-04 a las 14h45 del sitio web:



- <http://weather.msfc.nasa.gov/GOES/>.

Estas son imágenes del satélite interactivo global geostacionario del clima de la NASA.

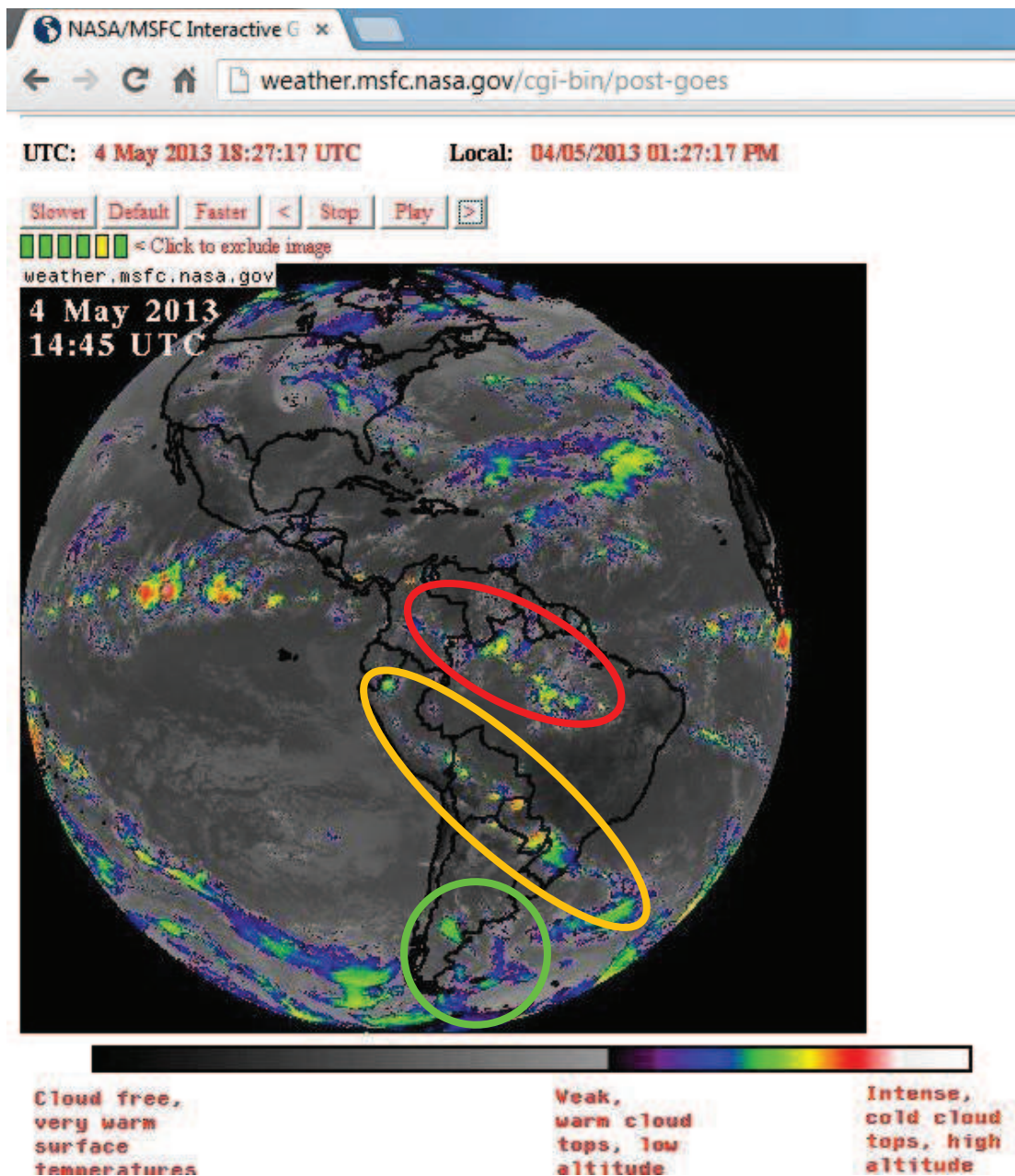


Fig. 3.18. Imagen satelital NASA para el continente americano obtenida a las 14h45 del 2013-05-04.

Para el análisis de esta primera imagen vamos a tomar en cuenta tres regiones con precipitación diferentes. La primera región abarca parte de Colombia,

Venezuela, hasta el centro de Brasil y está delimitada por el óvalo de color rojo. Según la simbología indicada en la parte inferior de la imagen tenemos un tipo de precipitación entre débil e intensa. La segunda región comprende el sur de Ecuador, pasando por Perú, Bolivia, Paraguay, el sur de Brasil y parte del Atlántico y está delimitada por el óvalo de color amarillo. En esta región tenemos un tipo de precipitación entre débil e intensa que se acentúa más en Paraguay. Y la tercera región incluye la parte sur de Chile, Argentina, el sur del Atlántico y está delimitada por el óvalo de color verde. Aquí tenemos un tipo de precipitación también entre débil e intensa.

La figura 3.19 corresponde a la imagen obtenida el 2013-05-04 a las 15h00 correspondiente al sitio web:

- [http://www.eltiempo24.es/imagenes\\_de\\_satelite/america\\_del\\_sur](http://www.eltiempo24.es/imagenes_de_satelite/america_del_sur)

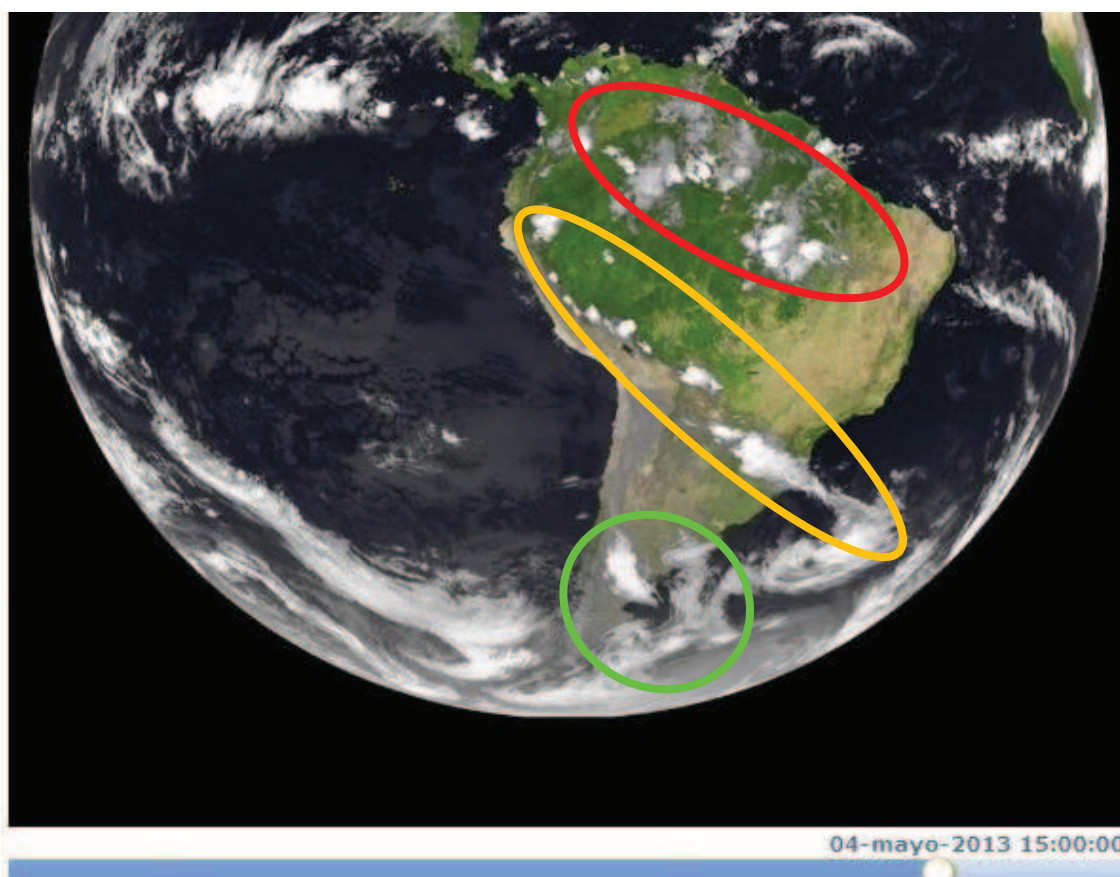


Fig. 3.19. Imagen satelital para el continente americano obtenida a las 15h00 del 2013-05-04.

En el análisis de esta segunda imagen se consideran también tres regiones con nubosidades diferentes. La primera región abarca el óvalo de color rojo y se aprecia cierto tipo de nubosidad normal. La segunda región comprende el óvalo de color amarillo y se aprecia una tendencia de nubosidad desde Ecuador, pasando por Perú, Bolivia, el sur de Brasil, acentuándose en Paraguay y parte del Atlántico. Y la tercera región está delimitada por el óvalo de color verde donde se aprecia nubosidad normal al sur de Chile, Argentina y parte del Atlántico.

La figura 3.20 corresponde a la imagen obtenida de la simulación el 2013-05-04 a las 15h01.

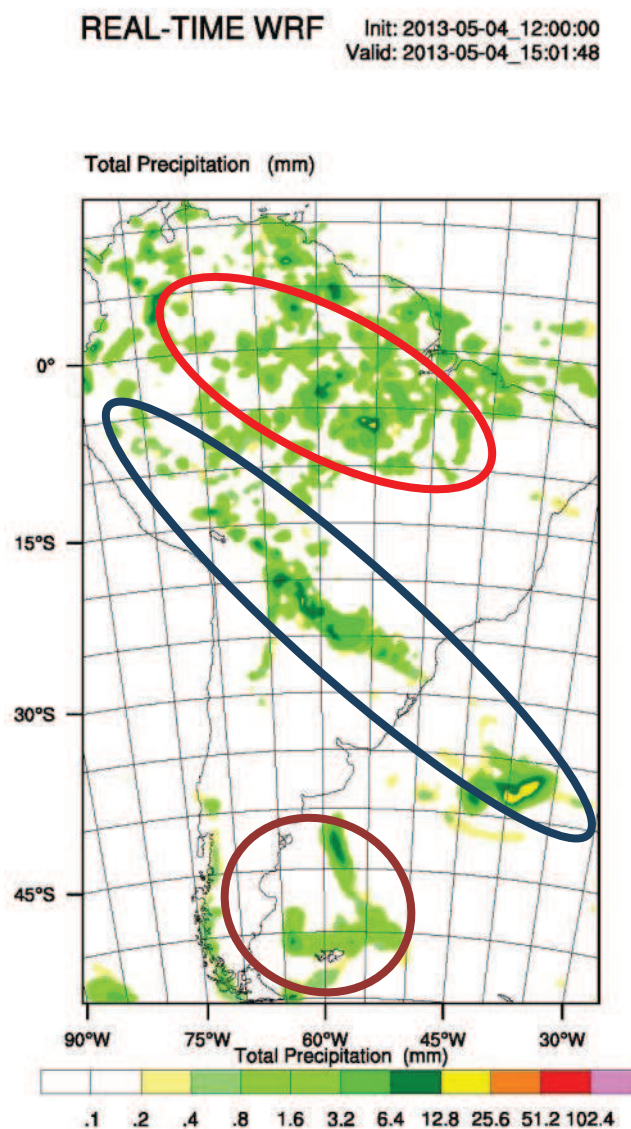


Fig. 3.20. Imagen de simulación obtenida a las 15h01 del 2013-05-04.

En la tercera imagen consideramos también tres regiones de análisis. En la primera región delimitada por el óvalo de color rojo tenemos niveles de precipitación entre 0.8 y 3.2 mm en la región de Colombia y Venezuela. En ciertos puntos de Brasil tenemos niveles entre 6.4 y 12.8 mm. En la segunda región correspondiente al óvalo de color azul tenemos precipitación entre 0.8 y 3.2 mm desde Ecuador, a lo largo de Perú y Bolivia. En Paraguay, el sur de Brasil y parte del Atlántico tenemos niveles que llegan hasta los 12.8 y 25.6 mm. En la tercera región demarcada por el óvalo de color marrón tenemos niveles entre 0.8 y 1.6 mm al sur de Chile, y niveles 1.6 y 12.8 mm en la parte sur del Atlántico.

Finalmente en la figura 3.21 se presentan en conjunto las tres imágenes mostradas anteriormente para verificar las similitudes entre la imagen obtenida en la simulación y las satelitales. Cabe señalar que de las tres regiones analizadas dentro de cada imagen, la región central que abarca Ecuador, atraviesa Perú, Bolivia, Paraguay, el sur de Brasil y parte del Atlántico, es la que más similitudes presenta en cuanto a precipitación. Además en la tercera región que comprende el sur del continente, la precipitación en el sector del Atlántico obtenida en la simulación, tienen un nivel de semejanza más cercana a la presentada por la imagen del satélite de la NASA.



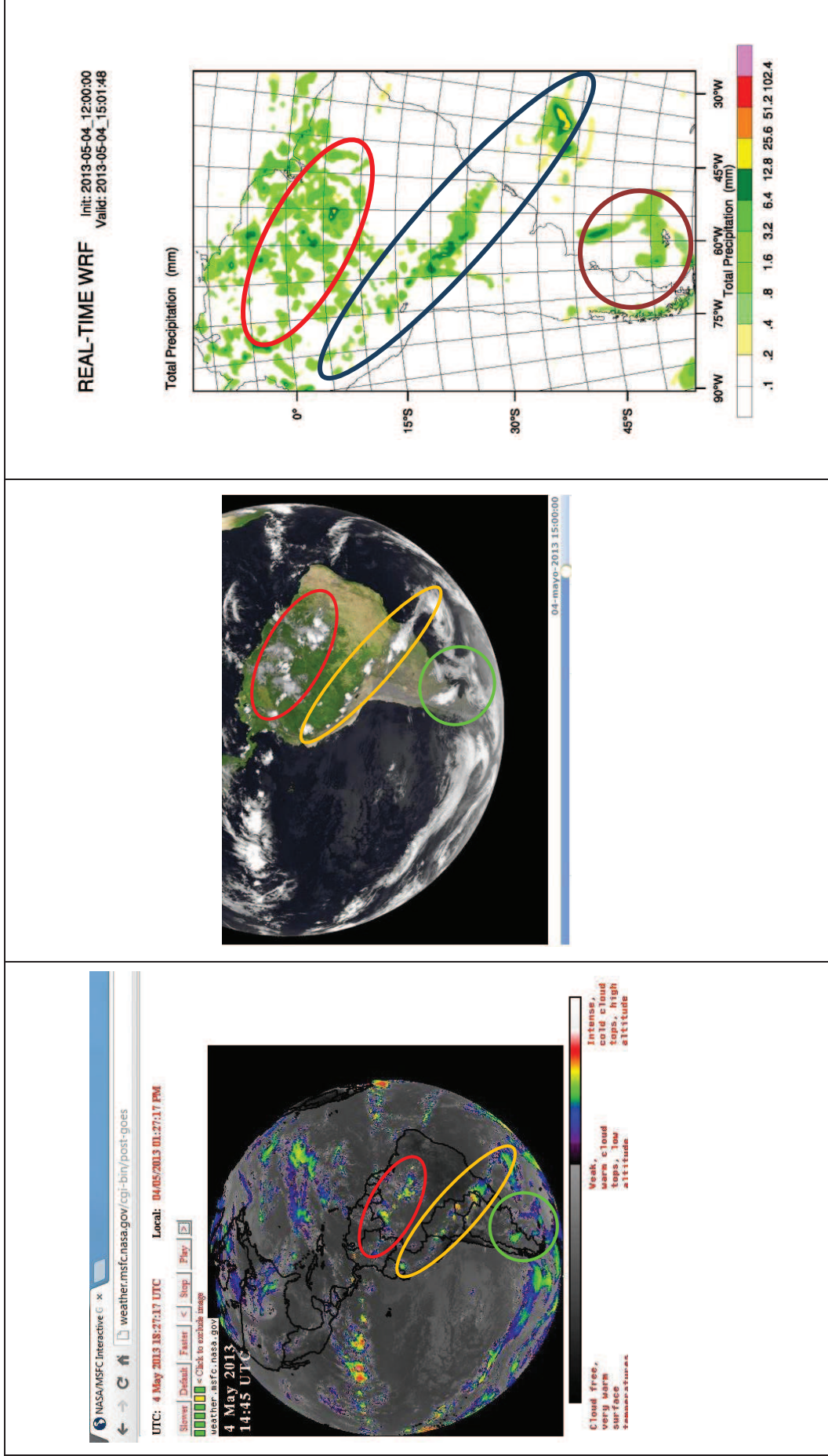


Fig. 3.21. Comparación entre dos imágenes satelitales y la imagen obtenida de la simulación.

## CAPÍTULO 4

### CONCLUSIONES Y RECOMENDACIONES

#### 4.1 CONCLUSIONES

La automatización del modelo *WRF* se completó de acuerdo al objetivo principal de este proyecto. Este objetivo se cumplió gracias al fundamento teórico presentado en el primer capítulo, en el cual se expone el funcionamiento de todos los componentes del modelo y las diferentes tareas que involucran la ejecución del mismo. De esta forma se pudieron diseñar procesos de entrada y salida de datos que se plasmaron en *scripts* de programación que automatizaron el funcionamiento del modelo, facilitando de esta forma la experiencia del usuario.

Luego de la fase de diseño de los procesos de entrada y salida de datos del modelo, la programación de *scripts* de automatización se facilitó gracias a la utilización de comandos empleados en tareas específicas: *wget* para la descarga de datos, y *sed* para la búsqueda y reemplazo de caracteres o cadenas de caracteres. Con *wget* se trabajó con la opción *wget -c* que resulta útil si por alguna razón no prevista la descarga se detiene, entonces se puede retomarla en el punto donde se quedó sin el riesgo de perder los datos descargados hasta antes de producirse el fallo. Con el comando *sed*, se busca cadenas de caracteres correspondientes a parámetros específicos dentro de los archivos de configuración *namelist.wps* y *namelist.input*, y se cambian de acuerdo a los valores necesarios para las simulaciones de este proyecto.

El tipo de datos meteorológicos con que se trabaja influyen directamente en el tiempo empleado en todo el proceso de automatización. En el caso específico de este proyecto, si se trabajan con datos formato *GFS* el proceso completo de simulación habría tomado alrededor de 12 horas, mientras que con los datos formato *FNL* el proceso duraría entre 3 y 4 horas. Esto se debe a que los datos

*GFS* son más pesados que los *FNL*, pues contienen información meteorológica más detallada.

Una de las desventajas de trabajar con los datos *FNL* radica en que durante algunas de las pruebas se presentan errores, al leer y descifrar los datos. En algunos casos no se encuentran algunos campos indispensables con sus respectivos valores. Para esto será necesario incluir datos de días anteriores para que suplan la falta de los valores no encontrados.

Uno de los problemas que se presenta en los países de Sudamérica es la escasez de observaciones de superficie y altura por no disponer de una red suficientemente densa de estaciones de medición. En estas circunstancias, el modelo *WRF* y sus pronósticos a corto plazo proporcionarán datos importantes de las condiciones atmosféricas de regiones donde la carencia de datos es una constante.

## 4.2 RECOMENDACIONES

Existen varias versiones de las librerías *netCDF* y del código fuente del modelo. Luego de haber realizado varias pruebas se llegó a establecer la compatibilidad entre la versión 4.1.2 de la librería *netCDF* y la versión 3.2.1 del código fuente de *WRF* y *WPS*, por lo que se recomienda el uso de estas versiones para la realización de cualquier tipo de pruebas.

En caso de implantar nuevamente todo el sistema se recomienda especial cuidado con la instalación de los paquetes básicos. En este caso la guía proporcionada en la página oficial de usuarios del modelo *WRF* menciona solamente los compiladores, lenguaje *Perl* y el paquete *netCDF*. Se recomienda seguir la sección de paquetes adicionales dentro de la configuración del servidor en el Capítulo 2. Dichos paquetes aparecieron como requisitos a medida que fueron realizadas pruebas antes de instalar la versión final del sistema.

Se recomienda que al trabajar con los datos iniciales meteorológicos *GFS* desde *NCEP*, se cuente con una infraestructura de comunicaciones robusta, que incluya una excelente velocidad para descarga de datos, de preferencia en el orden de decenas de Mbps. Además, luego de las pruebas realizadas, se pudo verificar que los datos meteorológicos correspondientes al mediodía están disponibles aproximadamente desde las 10:00 AM en la página de *NCEP*, lo cual se debe tomar en cuenta para iniciar el proceso de descarga de datos.

Dependiendo del tipo de datos meteorológicos con el que se trabaje para realizar las simulaciones, sean estos del tipo *GFS* o *FNL*, se recomienda tener especial cuidado con dos parámetros específicos: *fg\_name* y *prefix*, dentro del archivo de configuración *namelist.wps*. Se debe verificar que al trabajar con datos *GFS*, estos dos parámetros deben ser iguales a 'GFS2', y para el tipo de datos *FNL*, ambos deben ser iguales a 'FILE'. Esto resulta importante durante el proceso de descifrado de datos meteorológicos correspondiente al programa *ungrib.exe*, y de no cumplirse lo especificado anteriormente, el descifrado de datos no se realiza y todo el proceso de simulación falla. (Ver Anexo A).



Existen también otros datos para las condiciones iniciales que son proporcionados por el Centro de Modelado Científico de la Universidad de Zulia de Venezuela y que son utilizados para realización de predicciones numéricas por parte del INAHMI. La desventaja de acceder a estos datos radica en que tienen un costo y el objetivo de este proyecto es demostrar el funcionamiento del modelo *WRF* con herramientas libres y que estén al alcance de todo el público.

## CAPÍTULO 5

### BIBLIOGRAFÍA Y ANEXOS

#### 5.1 BIBLIOGRAFÍA

- [1] U.C.A.R. (2008). *Funcionamiento de los modelos de mesoescala*. [En línea]. Disponible: [http://www.meted.ucar.edu/mesoprim/models\\_es/](http://www.meted.ucar.edu/mesoprim/models_es/)
- [2] J. Dudhia. (1 de Febrero de 2010). *WRF Modeling System Overview*. [En línea]. Disponible: [http://www.mmm.ucar.edu/wrf/users/tutorial/201001/WRF\\_Overview\\_Dudhia.pdf](http://www.mmm.ucar.edu/wrf/users/tutorial/201001/WRF_Overview_Dudhia.pdf)
- [3] B. Gomez. (28 de Febrero de 2008). *Meteorological Models of MeteoGalicia*. [En línea]. Disponible: <http://www.project-easy.info/Biblioteca/Uploads/Tools/Models/Met-Models-EASY.pdf>
- [4] U.C.A.R. (4 de Junio de 2011). *WRF User Page*. [En línea]. Disponible: [http://www.mmm.ucar.edu/wrf/users/docs/user\\_guide\\_V3/contents.html](http://www.mmm.ucar.edu/wrf/users/docs/user_guide_V3/contents.html)
- [5] I. Gallai. (13 de Marzo de 2008). *A guide to the WRF numerical weather prediction model*. [En línea]. Disponible: [http://www.unitn.it/files/download/16654/foralps\\_tr\\_7.pdf](http://www.unitn.it/files/download/16654/foralps_tr_7.pdf)
- [6] U.C.A.R. (20 de Abril de 2011). *WRF ARW Online Tutorial*. [En línea]. Disponible: <http://www.mmm.ucar.edu/wrf/OnLineTutorial/Basics/index.html>

## 5.2 ANEXOS

### ANEXO A. ARCHIVO DE CONFIGURACIÓN: *namelist.wps*

```

&share
  wrf_core = 'ARW',
  max_dom = 3,
  start_date = '2010-09-13_00:00:00','2010-09-13_00:00:00','2010-09-
13_00:00:00',
  end_date   = '2010-09-15_00:00:00','2010-09-15_00:00:00','2010-09-
15_00:00:00',
  interval_seconds = 10800
  io_form_geogrid = 2,
  debug_level = 0,
  /
&geogrid
  parent_id          = 1,1,2,
  parent_grid_ratio = 1,3,3,
  i_parent_start     = 1,29,31,
  j_parent_start     = 1,28,16,
  e_we               = 100,115,226,
  e_sn               = 100,82,139,
  geog_data_res      = '10m','30s','30s',
  dx = 27000,
  dy = 27000,
  map_proj = 'lambert',
  ref_lat  = 39.516,
  ref_lon  = -4.906,
  truelat1 = 39.516,
  truelat2 = 39.516,
  stand_lon = -4.906,
  geog_data_path = '/home/wrf/WRF/data/geog'
  opt_geogrid_tbl_path = '/home/wrf/WRF/WPS/geogrid'
  ref_x = 50.0,
  ref_y = 50.0,
  /
&ungrib
  out_format = 'WPS',
  prefix = 'FILE',
  /
&metgrid
  fg_name = 'FILE'
  io_form_metgrid = 2,
  opt_metgrid_tbl_path = '/home/wrf/WRF/WPS/metgrid',
  /
&mod_levs
  press_pa = 201300 , 200100 , 100000 ,
            95000 , 90000 ,
            85000 , 80000 ,
            75000 , 70000 ,
            65000 , 60000 ,
            55000 , 50000 ,
            45000 , 40000 ,
            35000 , 30000 ,
            25000 , 20000 ,
            15000 , 10000 ,
            5000 , 1000
  /

```

**ANEXO B. ARCHIVO DE CONFIGURACIÓN: *namelist.input***

```

&time_control
run_days           = 0,
run_hours          = 48,
run_minutes        = 0,
run_seconds        = 0,
start_year         = 2010, 2010, 2010,
start_month        = 09, 09, 09,
start_day          = 13, 13, 13,
start_hour         = 00, 00, 00,
start_minute       = 00, 00, 00,
start_second       = 00, 00, 00,
end_year           = 2010, 2010, 2010,
end_month          = 09, 09, 09,
end_day            = 15, 15, 15,
end_hour           = 00, 00, 00,
end_minute         = 00, 00, 00,
end_second         = 00, 00, 00,
interval_seconds   = 10800
input_from_file    = .true., .true., .true.,
history_interval   = 60, 60, 60,
frames_per_outfile = 1000, 1000, 1000,
restart            = .false.,
restart_interval   = 5000,
io_form_history    = 2
io_form_restart    = 2
io_form_input      = 2
io_form_boundary   = 2
debug_level        = 0
/

&domains
time_step          = 180,
time_step_fract_num = 0,
time_step_fract_den = 1,
max_dom            = 3,
s_we               = 1,1,1,
e_we               = 100,115,226,
s_sn               = 1,1,1,
e_sn               = 100,82,139,
s_vert             = 1, 1, 1,
e_vert             = 28, 28, 28,
num_metgrid_levels = 27
dx                 = 27000,9000,3000,
dy                 = 27000,9000,3000,
grid_id            = 1,2,3,
parent_id          = 0,1,2,
i_parent_start     = 1,29,31,
j_parent_start     = 1,28,16,
parent_grid_ratio  = 1,3,3,
parent_time_step_ratio = 1,3,3,
feedback           = 1,
smooth_option      = 0
/

&physics
mp_physics         = 3, 3, 3,
ra_lw_physics      = 1, 1, 1,
ra_sw_physics      = 1, 1, 1,
radt               = 30, 30, 30,

```

```

sf_sfclay_physics      = 1,      1,      1,
sf_surface_physics    = 2,      2,      2,
bl_pbl_physics         = 1,      1,      1,
bldt                  = 0,      0,      0,
cu_physics             = 1,      1,      1,
cudt                  = 5,      5,      5,
isfflx                = 1,
ifsnow                 = 0,
icloud                = 1,
surface_input_source  = 1,
num_soil_layers        = 4,
sf_urban_physics      = 0,
maxiens                = 1,
maxens                 = 3,
maxens2                = 3,
maxens3                = 16,
ensdim                 = 144,
/
&fdda
/
&dynamics
w_damping              = 0,
diff_opt               = 1,
km_opt                 = 4,
diff_6th_opt           = 0,      0,      0,
diff_6th_factor        = 0.12,  0.12,  0.12,
base_temp               = 290.
damp_opt               = 0,
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,
/
&bdy_control
spec_bdy_width         = 5,
spec_zone              = 1,
relax_zone             = 4,
specified              = .true., .false., .false.,
nested                 = .false., .true., .true.,
/
&grib2
/
&namelist_quilt
nio_tasks_per_group = 0,
nio_groups = 1,
/

```

## ANEXO C. SCRIPT: GENERAL.SH

```

#!/bin/bash
#
# Script general de automatización del modelo WRF

```

```
#

# Iniciamos con la simulación para América del Sur

#cd /home/pancho/scripts02/scripts-americadelsur/

# Este proceso incluye la descarga de datos, cambios en los archivos
namelist, ejecución de los programas del bloque WPS y WRF, y finalmente
la publicación en el sitio web

/home/pancho/scripts02/scripts-americadelsur/1_descarga_datos.sh

/home/pancho/scripts02/scripts-americadelsur/2_cambiar_namelistwps_as.sh

/home/pancho/scripts02/scripts-americadelsur/3_correr_programaswps_as.sh

/home/pancho/scripts02/scripts-
americadelsur/4_cambiar_namelistinput_as.sh

/home/pancho/scripts02/scripts-americadelsur/5_correr_programaswrf_as.sh

/home/pancho/scripts02/scripts-americadelsur/6_graficar_as.sh

# Simulación para Ecuador

cd /home/pancho/scripts/scripts-ecuador/

# Este proceso incluye cambios en los archivos namelist, ejecución de
los programas del
# bloque WPS y WRF, y finalmente la publicación en el sitio web.
# No se incluye la descarga de datos que ya se realizó al inicio de la
anterior simulación

/home/pancho/scripts02/scripts-ecuador/1_cambiar_namelistwps_ec.sh

/home/pancho/scripts02/scripts-ecuador/2_correr_programaswps_ec.sh
```

```
/home/pancho/scripts02/scripts-ecuador/3_cambiar_namelistinput_ec.sh

/home/pancho/scripts02/scripts-ecuador/4_correr_programaswrf_ec.sh

/home/pancho/scripts02/scripts-ecuador/5_graficar_ec.sh

# Simulación para Pichincha

cd /home/pancho/scripts/scripts-pichincha/

# Este proceso incluye cambios en los archivos namelist, ejecución de
# los programas del bloque WPS y WRF, y finalmente la publicación en el
# sitio web.
# No se incluye la descarga de datos que ya se realizó al inicio de la
# anterior simulación

/home/pancho/scripts02/scripts-pichincha/1_cambiar_namelistwps_pi.sh

/home/pancho/scripts02/scripts-pichincha/2_correr_programaswps_pi.sh

/home/pancho/scripts02/scripts-pichincha/3_cambiar_namelistinput_pi.sh

/home/pancho/scripts02/scripts-pichincha/4_correr_programaswrf_pi.sh

/home/pancho/scripts02/scripts-pichincha/5_graficar_pi.sh
```

#### **ANEXO D. SCRIPT: 1\_DESCARGA-DATOS.SH**

```
#!/bin/bash

#

# Script para descarga y descifrado de datos iniciales del modelo WRF

#

# Iniciamos desde el directorio donde se almacenan los datos de entrada

cd /home/pancho/Desktop/wrf1/WRF/DATA/
```

```

# Creamos un directorio que lleva por nombre la fecha actual añadido el
# número 12 que hace referencia a la hora de los datos iniciales

mkdir `date +%Y%m%d`12

# Accedemos al directorio creado

cd `date +%Y%m%d`12

# Procedemos a la descarga de los datos iniciales para el modelo.
# Los datos se obtienen de la página de NCEP. Son datos globales de
acceso libre.

wget -c ftp://ftp.ncep.noaa.gov/pub/data/nccf/com/gfs/prod/gfs.`date
+%Y%m%d`12/gfs*

# Nos ubicamos en el directorio WPS para empezar con el proceso de
lectura
# y descifrado de los datos descargados.

cd /home/pancho/Desktop/wrf1/WRF/WPS/

# Creamos un enlace para la Vtable relevante y los archivos GRIB para
este conjunto de datos

ln -sf ungrib/Variable_Tables/Vtable.GFS Vtable
./link_gribcsh ../DATA/`date +%Y%m%d`12/gfs

```

## ANEXO E. SCRIPT: 2\_CAMBIAR\_NAMELISTWPS\_AS.SH

```

#!/bin/bash
#
# Script para modificar el archivo namelist.wps
#
# Nos ubicamos en el directorio donde se encuentra el archivo
namelist.wps

cd /home/pancho/Desktop/wrf1/WRF/WPS/

# Copiamos el contenido del archivo namelist.wps al archivo temporal

```



```

namelist.tmp
# para realizar los cambios de fechas y coordenadas

cp namelist.wps namelist.tmp

# Procedemos a cambiar fecha de inicio y fin de la simulación,
# e_we, e_sn, dx, dy, ref_lat, ref_lon, truelat, truelat2, stand_lon,
# prefix y fg_name

sed -e /start_date.*//start_date\ =\ "'`date +%Y-%m-
%d`_12:00:00`'",/g -e /end_date.*//end_date\ =\ "'`date --date='2
day' +%Y-%m-%d`_12:00:00`'",/g -e /e_we.*//e_we\ =\ 103,/g -e
/e_sn.*//e_sn\ =\ 159,/g -e /dx.*//dx\ =\ 50500,/g -e /dy.*//dy\
=\ 50500,/g -e /ref_lat.*//ref_lat\ =\ -21.989,/g -e
/ref_lon.*//ref_lon\ =\ -58.057,/g -e /truelat1.*//truelat1\ =\ -
21.989,/g -e /truelat2.*//truelat2\ =\ -21.989,/g -e
/stand_lon.*//stand_lon\ =\ -58.057,/g -e /prefix.*//prefix\ =\
"'GFS2'",/g -e /fg_name.*//fg_name\ =\ "'GFS2'",/g <namelist.tmp
>namelist.wps

rm namelist.tmp

```

### ANEXO F: 3\_CORRER\_PROGRAMASWPS\_AS.SH

```

#!/bin/bash
#
# Script para ejecutar los programas dentro del
# sistema de Pre-procesamiento WRF
#
# Nos ubicamos en el directorio WPS

cd /home/pancho/Desktop/wrf1/WRF/WPS/

# Procedemos a definir el dominio de simulación ejecutando el programa
geogrid

/home/pancho/Desktop/wrf1/WRF/WPS/geogrid.exe

# Describiremos los archivos GRIB2 para que puedan ser entendidos por el
modelo ejecutando el programa ungrib

/home/pancho/Desktop/wrf1/WRF/WPS/ungrib.exe >& ungrib_data.log

# Interpolamos los datos meteorológicos en el modelo ejecutando el
programa metgrid

/home/pancho/Desktop/wrf1/WRF/WPS/metgrid.exe

```

**ANEXO G. SCRIPT: 4\_CAMBIA\_NAMELISTINPUT\_AS.SH**

```
#!/bin/bash
#
# Script para modificar el archivo namelist.input
#
# Nos ubicamos en el directorio donde se encuentra el archivo
namelist.input

cd /home/pancho/Desktop/wrf1/WRF/WRFV3/test/em_real/

# Copiamos el contenido del archivo namelist.input al archivo temporal
namelist.tmp
# para realizar los cambios de fechas y coordenadas

cp namelist.input namelist.tmp

# Procedemos a cambiar fecha de inicio y fin de la simulación,
# history_interval, time_step, e_we, e_sn, dx y dy.

sed -e /run_days.*s//run_days\ =\ 0,/g -e /run_hours.*s//run_hours\ =\
48,/g -e /run_minutes.*s//run_minutes\ =\ 0,/g -e
/run_seconds.*s//run_seconds\ =\ 0,/g -e /start_year.*s//start_year\
=\ `date +%Y`,/g -e /start_month.*s//start_month\ =\ `date +%m`,/g -e
/start_day.*s//start_day\ =\ `date +%d`,/g -e
/start_hour.*s//start_hour\ =\ 12,/g -e
/start_minute.*s//start_minute\ =\ 00,/g -e
/start_second.*s//start_second\ =\ 00,/g -e /end_year.*s//end_year\ =\
`date +%Y`,/g -e /end_month.*s//end_month\ =\ `date +%m`,/g -e
/end_day.*s//end_day\ =\ `date --date='2 day' +%d`,/g -e
/end_hour.*s//end_hour\ =\ 12,/g -e /end_minute.*s//end_minute\ =\
00,/g -e /end_second.*s//end_second\ =\ 00,/g -e
/history_interval.*s//history_interval\ =\ 180,/g -e
/time_step.*s//time_step\ =\ 303,/g -e /e_we.*s//e_we\ =\ 103,/g -e
/e_sn.*s//e_sn\ =\ 159,/g -e /dx.*s//dx\ =\ 50500,/g -e /dy.*s//dy\
=\ 50500,/g <namelist.tmp >namelist.input

rm namelist.tmp
```

**ANEXO H. SCRIPT: 5\_CORRER\_PROGRAMASWRF\_AS.SH**

```
#!/bin/bash
#
# Script para ejecutar los programas finales para realizar
# la simulación
#
# Nos ubicamos en el directorio para correr el caso real

cd /home/pancho/Desktop/wrf1/WRF/WRFV3/test/em_real/
```

```

# Enlazamos los archivos met_em files creados con metgrid.exe

ln -sf /home/pancho/Desktop/wrf1/WRF/WPS/met_em.d01.* .

# Ejecutamos los programas real y wrf

/home/pancho/Desktop/wrf1/WRF/WRFV3/test/em_real/real.exe

/home/pancho/Desktop/wrf1/WRF/WRFV3/test/em_real/wrf.exe

# Añadimos la extensión .nc archivo final que ha sido creado para que
pueda ser
# reconocido por la herramienta gráfica NCL.
# Además añadimos _as_ para saber que es el archivo que corresponde a la
simulación de América del Sur

mv wrfout_d01_`date +%Y-%m-%d`_12:00:00 wrfout_d01_as_`date +%Y-%m-
%d`_12:00:00.nc

# Finalmente se enlaza el archivo final a la ubicación donde será
almacenado
# y mostrado en el servidor FTP

cd /home/pancho/Desktop/wrf1/ftpdata/

ln -sf
/home/pancho/Desktop/wrf1/WRF/WRFV3/test/em_real/wrfout_d01_as_`date
+%Y-%m-%d`_12:00:00.nc .

```

## ANEXO I. SCRIPT: 6\_GRAFICAR\_AS.SH

```

#!/bin/bash
#
# Script para mostrar los resultados de la simulación en formato .pdf
# utilizando la herramienta gráfica NCL
#
# Nos ubicamos en el directorio de la página web para que los archivos se
generen ahí

cd /home/pancho/Desktop/wrf1/Webserver/

# Copiamos el contenido del archivo wrf_Precip3.ncl al archivo temporal
wrf_Precip3.tmp para realizar cambios

cp wrf_Precip3.ncl wrf_Precip3.tmp

# Modificamos el archivo wrf_Precip3.ncl indicando la ubicación del
archivo de entrada .nc

sed -e /a\ =\ addfile.*//s//a\ =\
addfile\(\\"/home/pancho/Desktop/wrf1/WRF/WRFV3/test/em_real/wrfo
ut_d01_as_`date +%Y-%m-%d`_12:00:00.nc\", \"r\") -e /wks\ =\ .*//s//wks\

```

```

=\ gsn_open_wks\(type,\"AmericaDelSur\")/g <wrf_Precip3.tmp
>wrf_Precip3.ncl

rm wrf_Precip3.tmp

# Ejecutamos el programa NCL para generar el archivo AmericaDelSur.pdf
/home/pancho/ncl/bin/ncl wrf_Precip3.ncl

# Borramos archivos GRIB2, met_em
cd /home/pancho/Desktop/wrf1/WRF/WPS/
rm GFS2*
rm met_em.d01.*

cd /home/pancho/Desktop/wrf1/WRF/WRFV3/test/em_real/
rm met_em.d01.*

```

## ANEXO J. SCRIPT: 1\_CAMBIAR\_NAMELISTWPS\_EC.SH

```

#!/bin/bash
#
# Script para modificar el archivo namelist.wps
#
# Nos ubicamos en el directorio donde se encuentra el archivo
namelist.wps

cd /home/pancho/Desktop/wrf1/WRF/WPS/

# Copiamos el contenido del archivo namelist.wps al archivo temporal
namelist.tmp
# para realizar los cambios de fechas y coordenadas

cp namelist.wps namelist.tmp

# Procedemos a cambiar fecha de inicio y fin de la simulación,
# e_we, e_sn, dx, dy, ref_lat, ref_lon, truelat, truelat2, stand_lon,
# prefix y fg_name

sed -e /e_we.*s//e_we\ =\ 100,/g -e /e_sn.*s//e_sn\ =\ 98,/g -e
/dx.*s//dx\ =\ 10000,/g -e /dy.*s//dy\ =\ 10000,/g -e
/ref_lat.*s//ref_lat\ =\ -1.991,/g -e /ref_lon.*s//ref_lon\ =\ -
78.321,/g -e /truelat1.*s//truelat1\ =\ -1.991,/g -e
/truelat2.*s//truelat2\ =\ -1.991,/g -e /stand_lon.*s//stand_lon\ =\ -
78.321,/g -e /prefix.*s//prefix\ =\ ""GFS2"",/g -e
/fg_name.*s//fg_name\ =\ ""GFS2"",/g <namelist.tmp >namelist.wps

rm namelist.tmp

```

## ANEXO K: 2\_CORRER\_PROGRAMASWPS\_EC.SH

```
#!/bin/bash
#
# Script ejecutar los programas dentro del
# sistema de Pre-procesamiento WRF
#
# Nos ubicamos en el directorio WPS
cd /home/pancho/Desktop/wrf1/WRF/WPS/

# Procedemos a definir el dominio de simulación ejecutando el programa
geogrid

/home/pancho/Desktop/wrf1/WRF/WPS/geogrid.exe

# Describimos los archivos GRIB2 para que puedan ser entendidos por el
modelo ejecutando el programa ungrib

/home/pancho/Desktop/wrf1/WRF/WPS/ungrib.exe >& ungrib_data.log

# Interpolamos los datos meteorológicos en el modelo ejecutando el
programa metgrid

/home/pancho/Desktop/wrf1/WRF/WPS/metgrid.exe
```

## ANEXO L. SCRIPT: 3\_CAMBIA\_NAMELISTINPUT\_EC.SH

```
#!/bin/bash
#
# Script para modificar el archivo namelist.input
#
# Nos ubicamos en el directorio donde se encuentra el archivo
namelist.input
cd /home/pancho/Desktop/wrf1/WRF/WRFV3/test/em_real/

# Copiamos el contenido del archivo namelist.input al archivo temporal
namelist.tmp
# para realizar los cambios de fechas y coordenadas
cp namelist.input namelist.tmp

# Procedemos a cambiar fecha de inicio y fin de la simulación,
# history_interval, time_step, e_we, e_sn, dx y dy.
sed -e /history_interval.*/s//history_interval\ =\ 180,/g -e /time_step\
```

```

./s//time_step\ =\ 43,/g -e /e_we.*/s//e_we\ =\ 100,/g -e
/e_sn.*/s//e_sn\ =\ 98,/g -e /dx\ .*/s//dx\ =\ 10000,/g -e /dy\
.*/s//dy\ =\ 10000,/g <namelist.tmp >namelist.input

rm namelist.tmp

```

## ANEXO M. SCRIPT: 4\_CORRER\_PROGRAMASWRF\_EC.SH

```

#!/bin/bash
#
# Script para ejecutar los programas finales para realizar
# la simulación
#
# Nos ubicamos en el directorio para correr el caso real
cd /home/pancho/Desktop/wrf1/WRF/WRFV3/test/em_real/

# Enlazamos los archivos met_em files creados con metgrid.exe
ln -sf /home/pancho/Desktop/wrf1/WRF/WPS/met_em.d01.* .

# Ejecutamos los programas real y wrf
/home/pancho/Desktop/wrf1/WRF/WRFV3/test/em_real/real.exe

/home/pancho/Desktop/wrf1/WRF/WRFV3/test/em_real/wrf.exe

# Añadimos la extensión .nc archivo final que ha sido creado para que
# pueda ser
# reconocido por la herramienta gráfica NCL.
# Además añadimos _as_ para saber que es el archivo que corresponde a la
# simulación de Ecuador

mv wrfout_d01_`date +%Y-%m-%d`_12:00:00 wrfout_d01_ec_`date +%Y-%m-
%d`_12:00:00.nc

# Finalmente se enlaza el archivo final a la ubicación donde será
# almacenado
# y mostrado en el servidor FTP

cd /home/pancho/Desktop/wrf1/ftpdata/

ln -sf
/home/pancho/Desktop/wrf1/WRF/WRFV3/test/em_real/wrfout_d01_ec_`date
+%Y-%m-%d`_12:00:00.nc .

```

## ANEXO N. SCRIPT: 5\_GRAFICAR\_EC.SH

```

#!/bin/bash
#

```

```

# Script para mostrar los resultados de la simulación en formato .pdf
# utilizando la herramienta gráfica NCL
#

# Nos ubicamos en el directorio de la página web para que los archivos se
generen ahí

cd /home/pancho/Desktop/wrf1/Webserver/

# Copiamos el contenido del archivo wrf_Precip3.ncl al archivo temporal
wrf_Precip3.tmp para realizar cambios

cp wrf_Precip3.ncl wrf_Precip3.tmp

# Modificamos el archivo wrf_Precip3.ncl indicando la ubicación del
archivo de entrada .nc

sed -e /a\ =\ addfile.*//s//a\ =\
addfile\(\\"/home/pancho/Desktop/wrf1/WRF/WRFV3/test/em_real/wrfo
ut_d01_ec_date +%Y-%m-%d`_12:00:00.nc\", \"r\") -e /wks\ =\ .*//s//wks\
=\ gsn_open_wks\ (type, \"Ecuador\")/g <wrf_Precip3.tmp >wrf_Precip3.ncl

rm wrf_Precip3.tmp

# Ejecutamos el programa NCL para generar el archivo Ecuador.pdf

/home/pancho/ncl/bin/ncl wrf_Precip3.ncl

# Borramos archivos GRIB2, met_em

cd /home/pancho/Desktop/wrf1/WRF/WPS/

rm GFS2*

rm met_em.d01.*

cd /home/pancho/Desktop/wrf1/WRF/WRFV3/test/em_real/

rm met_em.d01.*

```

## ANEXO O. SCRIPT: 1\_CAMBIAR\_NAMELISTWPS\_PL.SH

```

#!/bin/bash
#
# Script para modificar el archivo namelist.wps
#

# Nos ubicamos en el directorio donde se encuentra el archivo
namelist.wps

cd /home/pancho/Desktop/wrf1/WRF/WPS/

```

```

# Copiamos el contenido del archivo namelist.wps al archivo temporal
namelist.tmp
# para realizar los cambios de fechas y coordenadas

cp namelist.wps namelist.tmp

# Procedemos a cambiar fecha de inicio y fin de la simulación,
# e_we, e_sn, dx, dy, ref_lat, ref_lon, truelat, truelat2, stand_lon,
# prefix y fg_name

sed -e /e_we.*s//e_we\ =\ 127,/g -e /e_sn.*s//e_sn\ =\ 106,/g -e
/dx.*s//dx\ =\ 2033.333,/g -e /dy.*s//dy\ =\ 2033.333,/g -e
/ref_lat.*s//ref_lat\ =\ -1.058,/g -e /ref_lon.*s//ref_lon\ =\ -
78.055,/g -e /truelat1.*s//truelat1\ =\ -1.058,/g -e
/truelat2.*s//truelat2\ =\ -1.058,/g -e /stand_lon.*s//stand_lon\ =\ -
78.055,/g -e /prefix.*s//prefix\ =\ ""GFS2"",/g -e
/fg_name.*s//fg_name\ =\ ""GFS2"",/g <namelist.tmp >namelist.wps

rm namelist.tmp

```

## ANEXO P: 2\_CORRER\_PROGRAMASWPS\_PLSH

```

#!/bin/bash
#
# Script para ejecutar los programas dentro del
# sistema de Pre-procesamiento WRF
#
# Nos ubicamos en el directorio WPS
cd /home/pancho/Desktop/wrf1/WRF/WPS/

# Procedemos a definir el dominio de simulación ejecutando el programa
geogrid

/home/pancho/Desktop/wrf1/WRF/WPS/geogrid.exe

# Describimos los archivos GRIB2 para que puedan ser entendidos por el
modelo ejecutando el programa ungrib

/home/pancho/Desktop/wrf1/WRF/WPS/ungrib.exe >& ungrib_data.log

# Interpolamos los datos meteorológicos en el modelo ejecutando el
programa metgrid

/home/pancho/Desktop/wrf1/WRF/WPS/metgrid.exe

```



**ANEXO Q. SCRIPT: 3\_CAMBIA\_NAMELISTINPUT\_PL.SH**

```
#!/bin/bash
#
# Script para modificar el archivo namelist.input
#
# Nos ubicamos en el directorio donde se encuentra el archivo
namelist.input

cd /home/pancho/Desktop/wrf1/WRF/WRFV3/test/em_real/

# Copiamos el contenido del archivo namelist.input al archivo temporal
namelist.tmp
# para realizar los cambios de fechas y coordenadas

cp namelist.input namelist.tmp

# Procedemos a cambiar fecha de inicio y fin de la simulación,
# history_interval, time_step, e_we, e_sn, dx y dy.

sed -e /history_interval.*/s//history_interval\ =\ 60,/g -e /time_step\
.*/s//time_step\ =\ 37,/g -e /e_we.*/s//e_we\ =\ 127,/g -e
/e_sn.*/s//e_sn\ =\ 106,/g -e /dx\ .*/s//dx\ =\ 2033.333,/g -e /dy\
.*/s//dy\ =\ 2033.333,/g <namelist.tmp >namelist.input

rm namelist.tmp
```

**ANEXO R. SCRIPT: 4\_CORRER\_PROGRAMASWRF\_PL.SH**

```
#!/bin/bash
#
# Script para ejecutar los programas finales para realizar
# la simulación
#
# Nos ubicamos en el directorio para correr el caso real

cd /home/pancho/Desktop/wrf1/WRF/WRFV3/test/em_real/

# Enlazamos los archivos met_em files creados con metgrid.exe

ln -sf /home/pancho/Desktop/wrf1/WRF/WPS/met_em.d01.* .

# Ejecutamos los programas real y wrf

/home/pancho/Desktop/wrf1/WRF/WRFV3/test/em_real/real.exe

/home/pancho/Desktop/wrf1/WRF/WRFV3/test/em_real/wrf.exe
```

```

# Añadimos la extensión .nc archivo final que ha sido creado para que
pueda ser
# reconocido por la herramienta gráfica NCL.
# Además añadimos _as_ para saber que es el archivo que corresponde a la
simulación de Pichincha

mv wrfout_d01_`date +%Y-%m-%d`_12:00:00 wrfout_d01_pi_`date +%Y-%m-
%d`_12:00:00.nc

# Finalmente se enlaza el archivo final a la ubicación donde será
almacenado
# y mostrado en el servidor FTP

cd /home/pancho/Desktop/wrf1/ftpdata/

ln -sf
/home/pancho/Desktop/wrf1/WRF/WRFV3/test/em_real/wrfout_d01_pi_`date
+%Y-%m-%d`_12:00:00.nc .

```

## ANEXO S. SCRIPT: 5\_GRAFICAR\_PL.SH

```

#!/bin/bash
#
# Script para mostrar los resultados de la simulación en formato .pdf
# utilizando la herramienta gráfica NCL
#
# Nos ubicamos en el directorio de la página web para que los archivos se
generen ahí

cd /home/pancho/Desktop/wrf1/Webserver/

# Copiamos el contenido del archivo wrf_Precip3.ncl al archivo temporal
wrf_Precip3.tmp para realizar cambios

cp wrf_Precip3.ncl wrf_Precip3.tmp

# Modificamos el archivo wrf_Precip3.ncl indicando la ubicación del
archivo de entrada .nc

sed -e /a\ =\ addfile.*\/s\/a\ =\
addfile\(\\"\/home\/pancho\/Desktop\/wrf1\/WRF\/WRFV3\/test\/em_real\/wrfout_d01_pi_`date +%Y-%m-%d`_12:00:00.nc\", \"r\"\) -e /wks\ =\ .*\/s\/wks\
=\ gsn_open_wks\ (type, \"Pichincha\") /g <wrf_Precip3.tmp >wrf_Precip3.ncl

rm wrf_Precip3.tmp

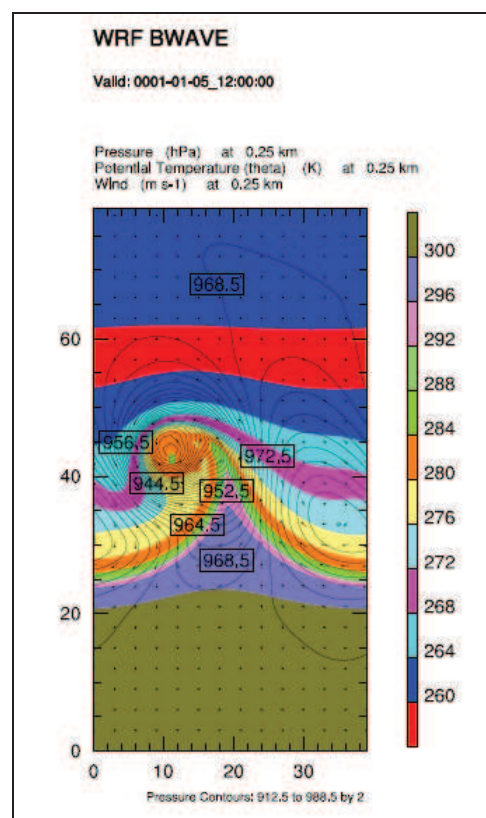
# Ejecutamos el programa NCL para generar el archivo Pichincha.pdf

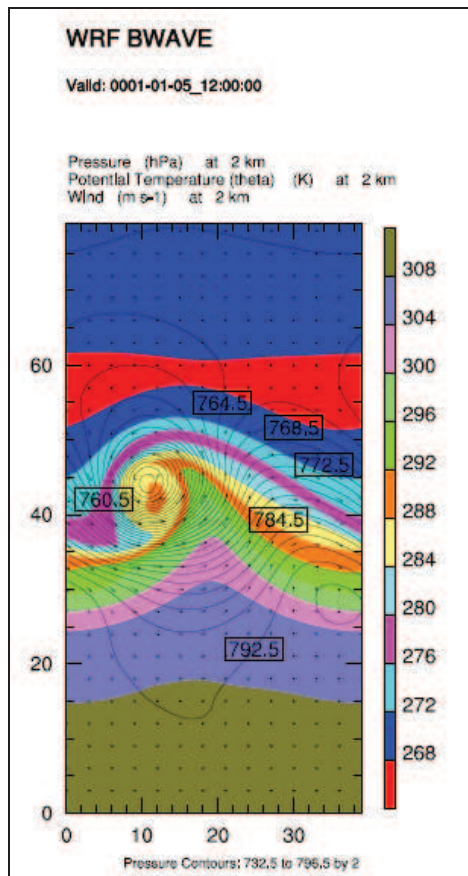
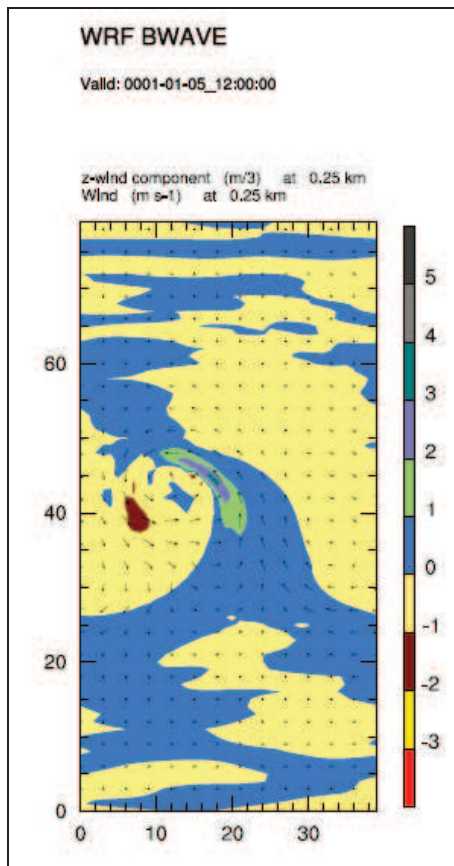
/home/pancho/ncl/bin/ncl wrf_Precip3.ncl

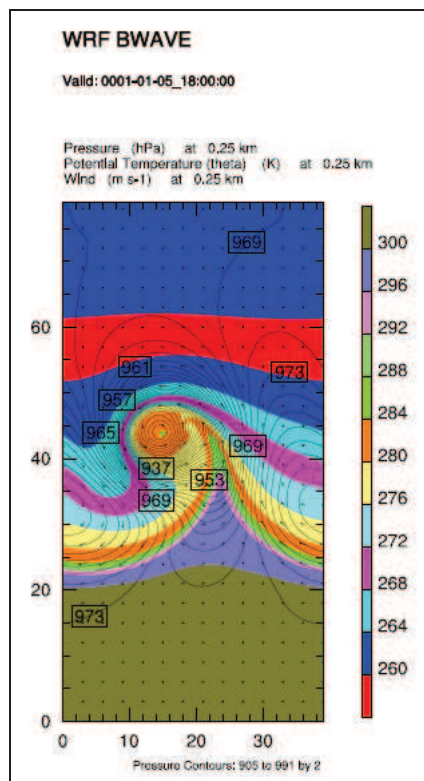
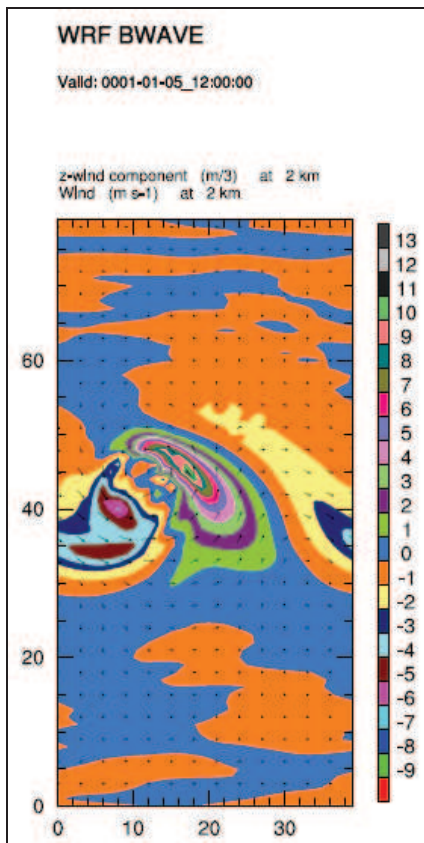
```

```
# Borrarnos archivos GRIB2, met_em  
cd /home/pancho/Desktop/wrf1/WRF/WPS/  
  
rm GFS2*  
  
rm met_em.d01.*  
  
cd /home/pancho/Desktop/wrf1/WRF/WRFV3/test/em_real/  
  
rm met_em.d01.*
```

## ANEXO T. GRÁFICAS DE SIMULACIÓN CASO IDEAL.

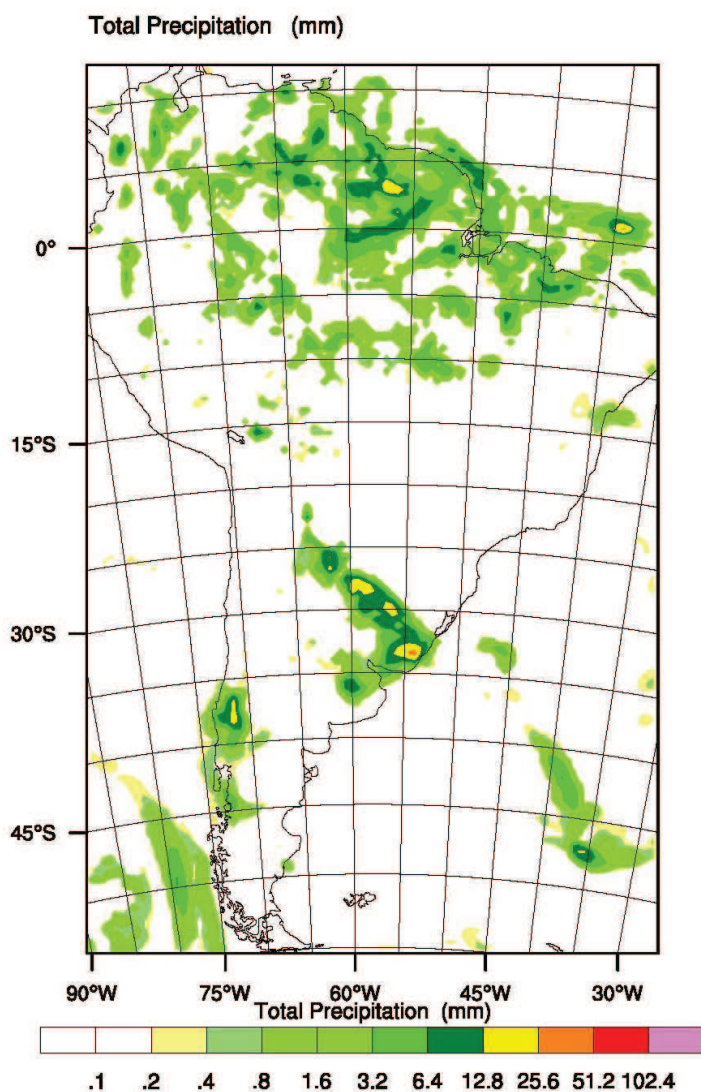






## ANEXO U. GRÁFICAS DE SIMULACIÓN PARA AMÉRICA DEL SUR.

**REAL-TIME WRF**    Init: 2013-05-03\_12:00:00  
Valid: 2013-05-03\_15:01:48

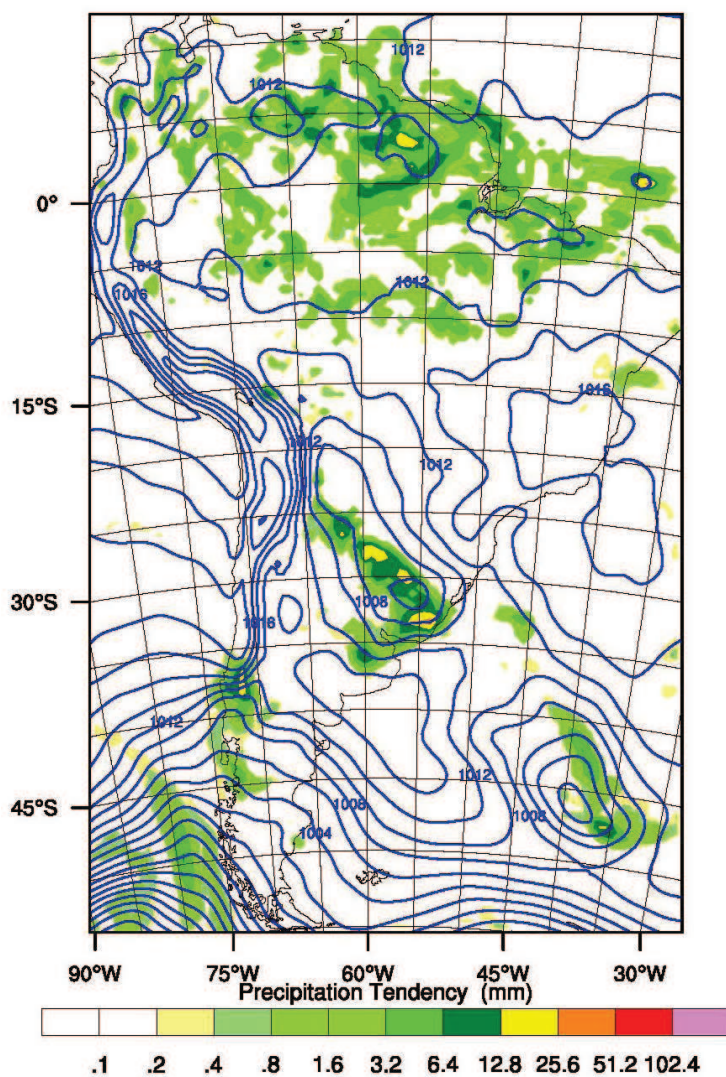


OUTPUT FROM WRF V3.2.1 MODEL  
WE = 103 ; SN = 159 ; Levels = 28 ; Dis = 50.5km ; Phys Opt = 3 ; PBL Opt = 1 ; Cu Opt = 1



**REAL-TIME WRF**    Init: 2013-05-03\_12:00:00  
Valid: 2013-05-03\_15:01:48

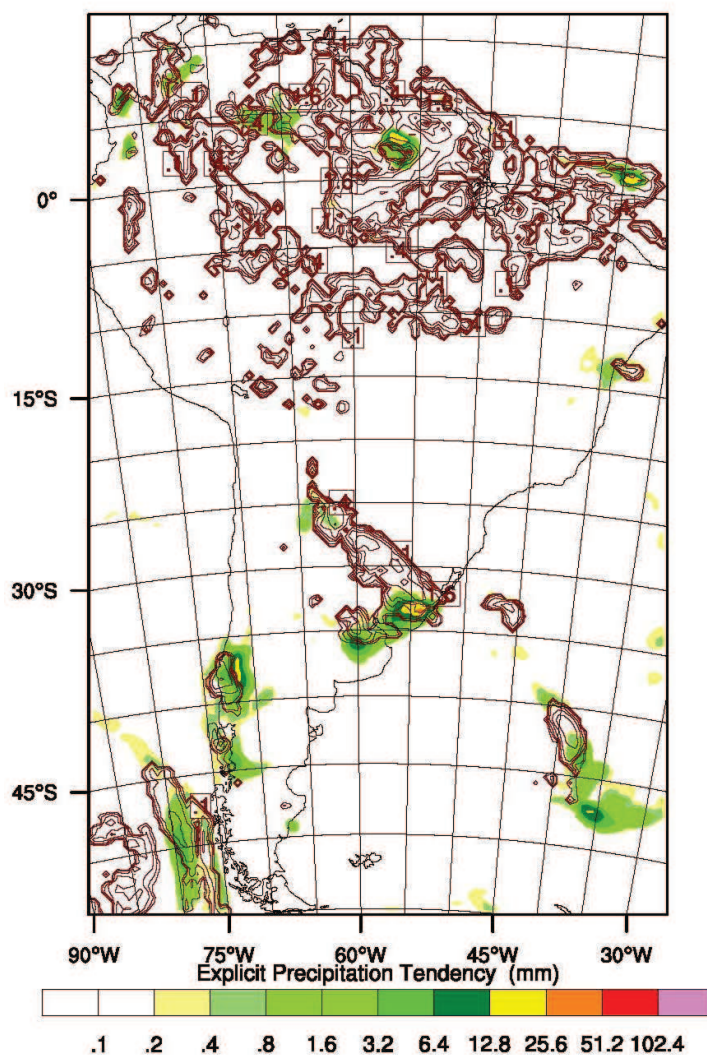
Precipitation Tendency from 2013-05-03\_12:00:00 to 2013-05-03\_15:01:48 (mm)  
Sea Level Pressure (hPa)



OUTPUT FROM WRF V3.2.1 MODEL  
WE = 103 ; SN = 159 ; Levels = 28 ; Dis = 50.5km ; Phys Opt = 3 ; PBL Opt = 1 ; Cu Opt = 1

**REAL-TIME WRF**    Init: 2013-05-03\_12:00:00  
Valid: 2013-05-03\_15:01:48

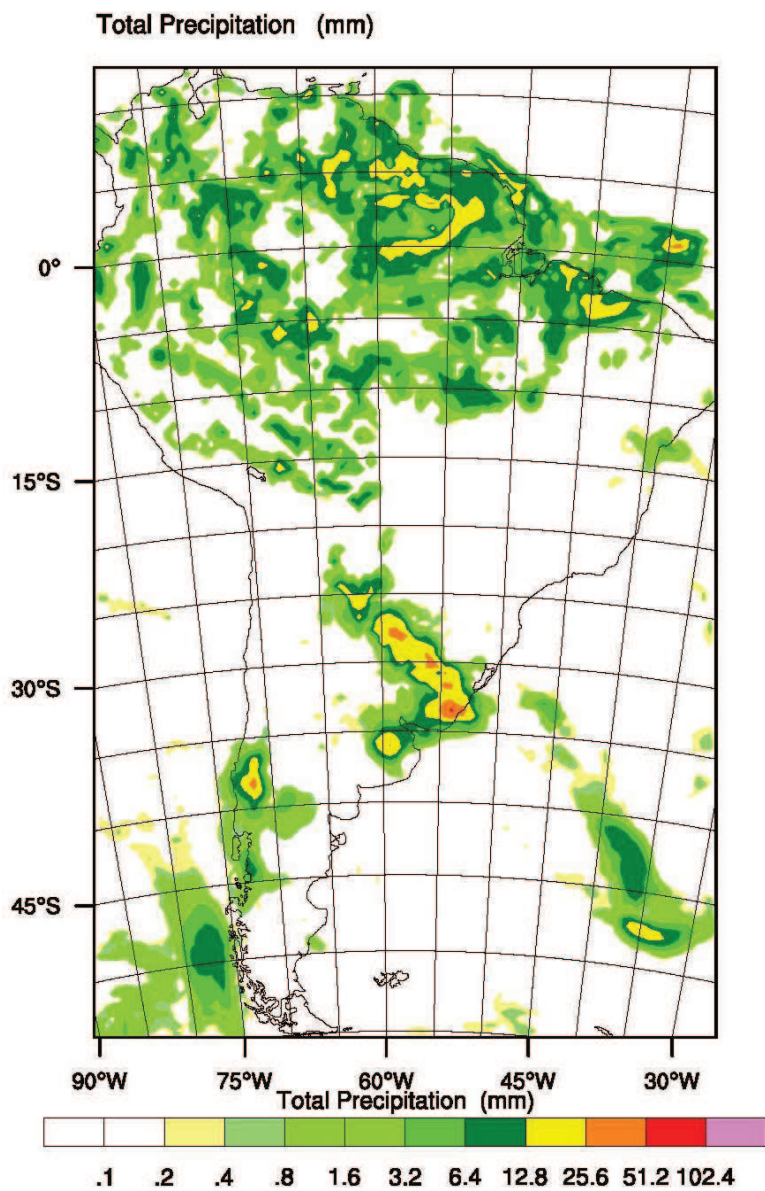
Explicit Precipitation Tendency from 2013-05-03\_12:00:00 to 2013-05-03\_15:01:48 (mm)  
Param Precipitation Tendency from 2013-05-03\_12:00:00 to 2013-05-03\_15:01:48 (mm)



OUTPUT FROM WRF V3.2.1 MODEL  
WE = 103 ; SN = 159 ; Levels = 28 ; Dis = 50.5km ; Phys Opt = 3 ; PBL Opt = 1 ; Cu Opt = 1



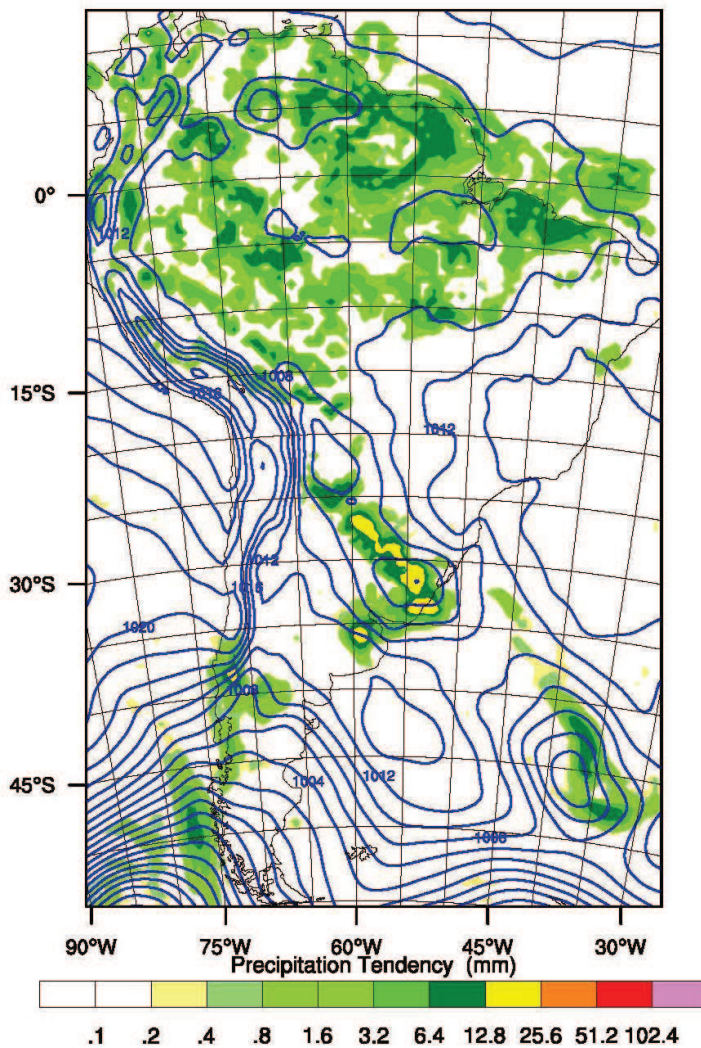
**REAL-TIME WRF**    Init: 2013-05-03\_12:00:00  
 Valid: 2013-05-03\_18:03:36



OUTPUT FROM WRF V3.2.1 MODEL  
 WE = 103 ; SN = 159 ; Levels = 28 ; Dis = 50.5km ; Phys Opt = 3 ; PBL Opt = 1 ; Cu Opt = 1

**REAL-TIME WRF**    Init: 2013-05-03\_12:00:00  
 Valid: 2013-05-03\_18:03:36

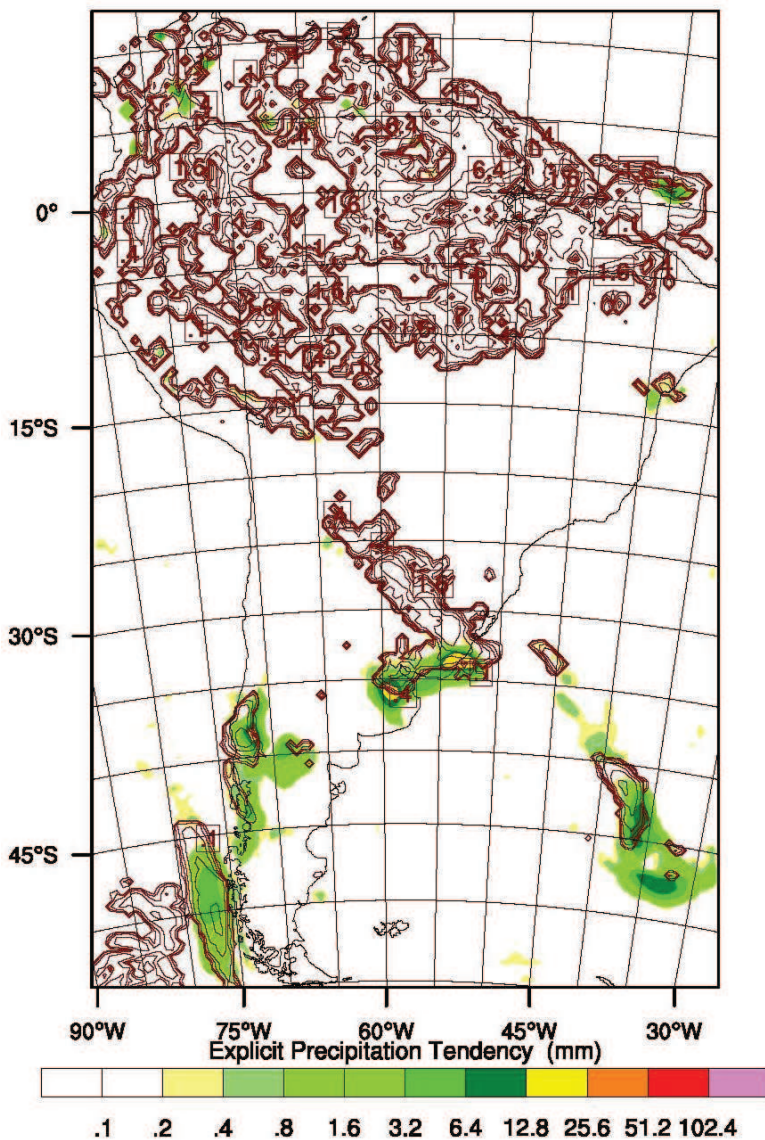
Precipitation Tendency from 2013-05-03\_15:01:48 to 2013-05-03\_18:03:36 (mm)  
 Sea Level Pressure (hPa)



OUTPUT FROM WRF V3.2.1 MODEL  
 WE = 103 ; SN = 159 ; Levels = 28 ; Dis = 50.5km ; Phys Opt = 3 ; PBL Opt = 1 ; Cu Opt = 1

**REAL-TIME WRF**    Init: 2013-05-03\_12:00:00  
 Valid: 2013-05-03\_18:03:36

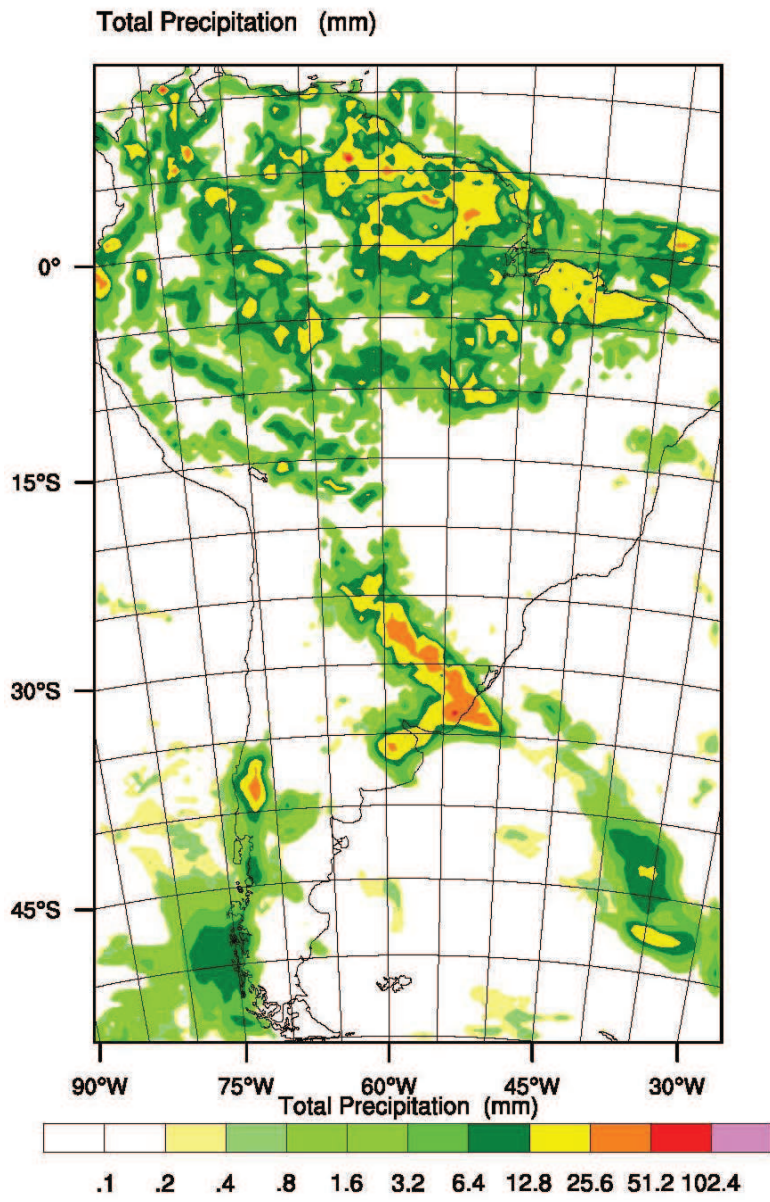
Explicit Precipitation Tendency from 2013-05-03\_15:01:48 to 2013-05-03\_18:03:36 (mm)  
 Param Precipitation Tendency from 2013-05-03\_15:01:48 to 2013-05-03\_18:03:36 (mm)



OUTPUT FROM WRF V3.2.1 MODEL  
 WE = 103 ; SN = 159 ; Levels = 28 ; Dis = 50.5km ; Phys Opt = 3 ; PBL Opt = 1 ; Cu Opt = 1



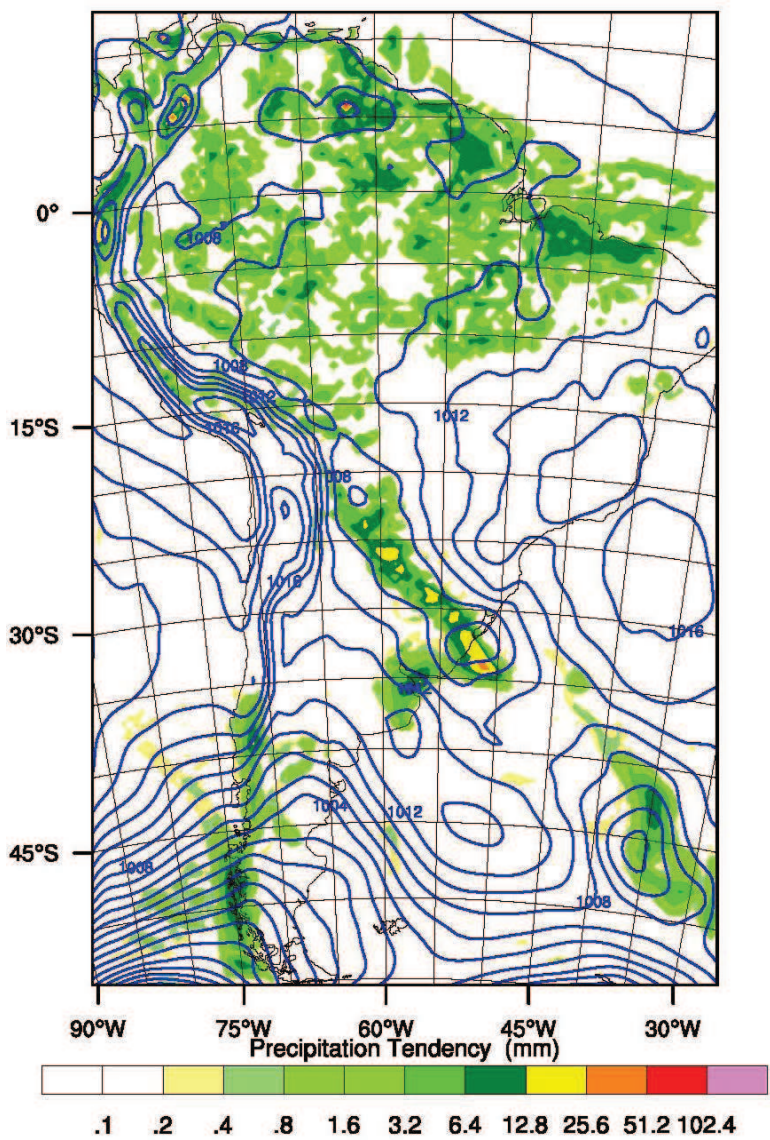
**REAL-TIME WRF**    Init: 2013-05-03\_12:00:00  
 Valid: 2013-05-03\_21:05:24



**OUTPUT FROM WRF V3.2.1 MODEL**  
 WE = 103 ; SN = 159 ; Levels = 28 ; Dis = 50.5km ; Phys Opt = 3 ; PBL Opt = 1 ; Cu Opt = 1

**REAL-TIME WRF**    Init: 2013-05-03\_12:00:00  
 Valid: 2013-05-03\_21:05:24

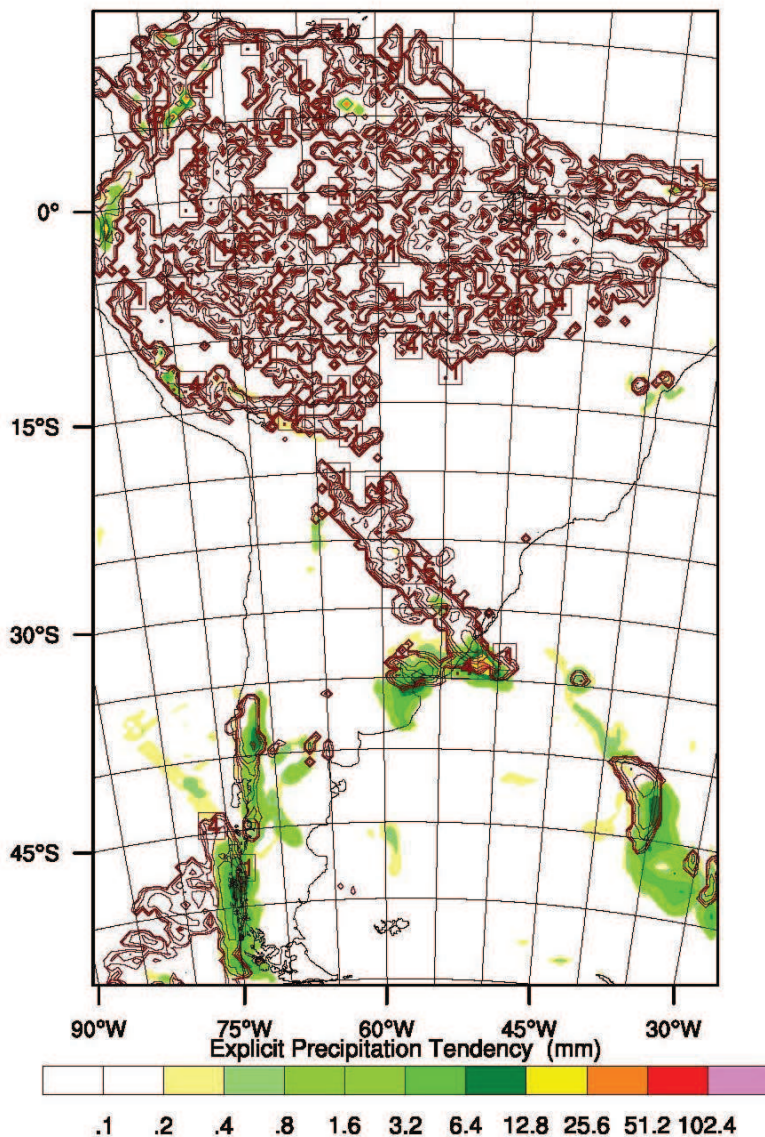
Precipitation Tendency from 2013-05-03\_18:03:36 to 2013-05-03\_21:05:24 (mm)  
 Sea Level Pressure (hPa)



OUTPUT FROM WRF V3.2.1 MODEL  
 WE = 103 ; SN = 159 ; Levels = 28 ; Dis = 50.5km ; Phys Opt = 3 ; PBL Opt = 1 ; Cu Opt = 1

**REAL-TIME WRF**    Init: 2013-05-03\_12:00:00  
Valid: 2013-05-03\_21:05:24

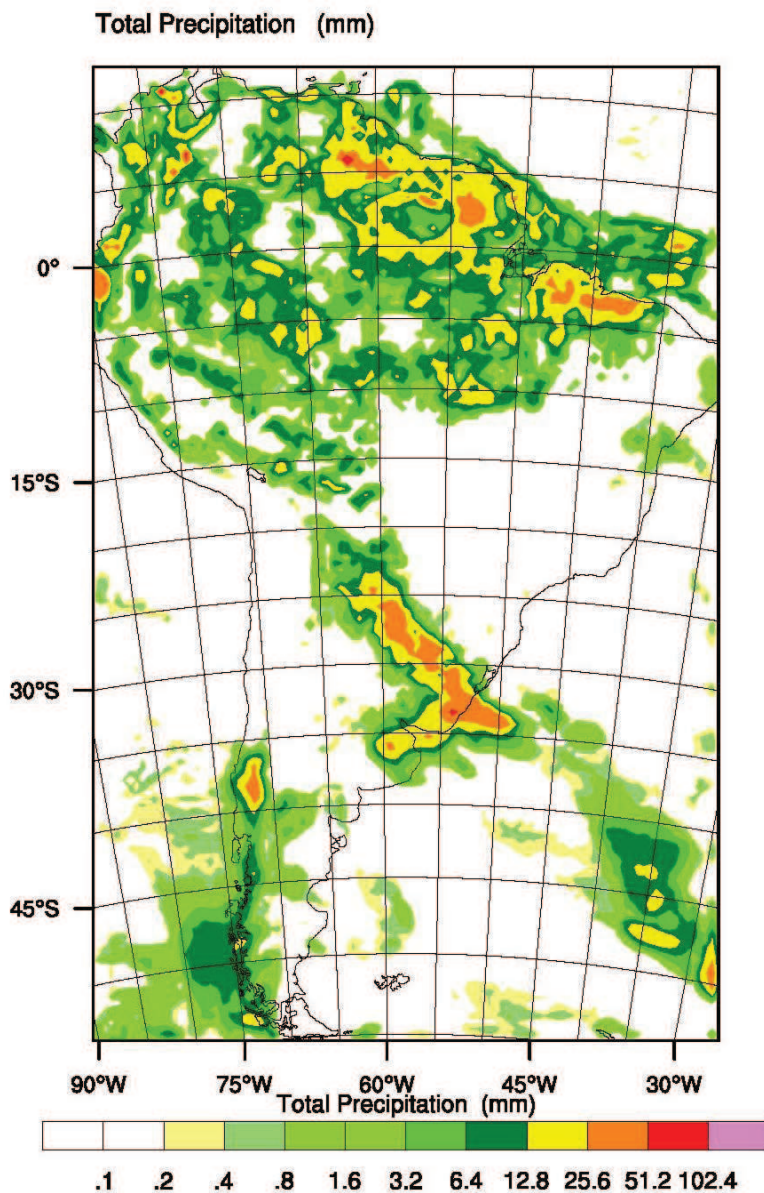
Explicit Precipitation Tendency from 2013-05-03\_18:03:36 to 2013-05-03\_21:05:24 (mm)  
Param Precipitation Tendency from 2013-05-03\_18:03:36 to 2013-05-03\_21:05:24 (mm)



OUTPUT FROM WRF V3.2.1 MODEL  
WE = 103 ; SN = 159 ; Levels = 28 ; Dis = 50.5km ; Phys Opt = 3 ; PBL Opt = 1 ; Cu Opt = 1



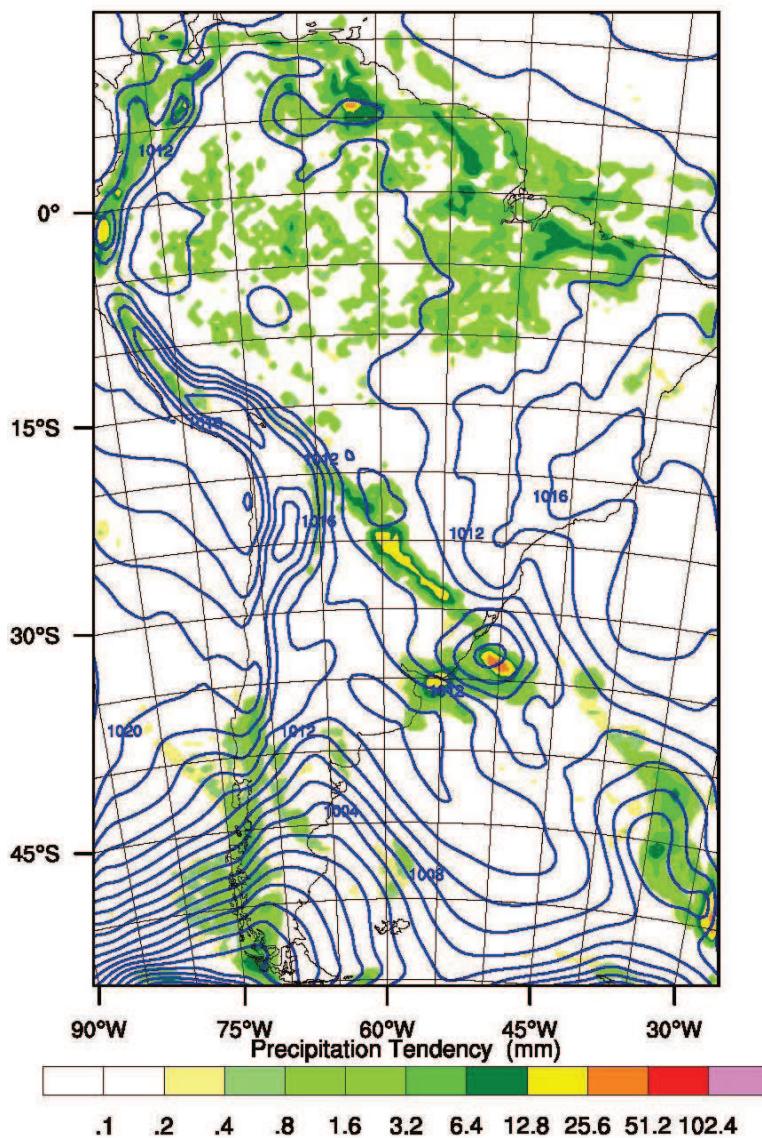
**REAL-TIME WRF**    Init: 2013-05-03\_12:00:00  
 Valid: 2013-05-04\_00:07:12



OUTPUT FROM WRF V3.2.1 MODEL  
 WE = 103 ; SN = 159 ; Levels = 28 ; Dis = 50.5km ; Phys Opt = 3 ; PBL Opt = 1 ; Cu Opt = 1

**REAL-TIME WRF**    Init: 2013-05-03\_12:00:00  
 Valid: 2013-05-04\_00:07:12

Precipitation Tendency from 2013-05-03\_21:05:24 to 2013-05-04\_00:07:12 (mm)  
 Sea Level Pressure (hPa)

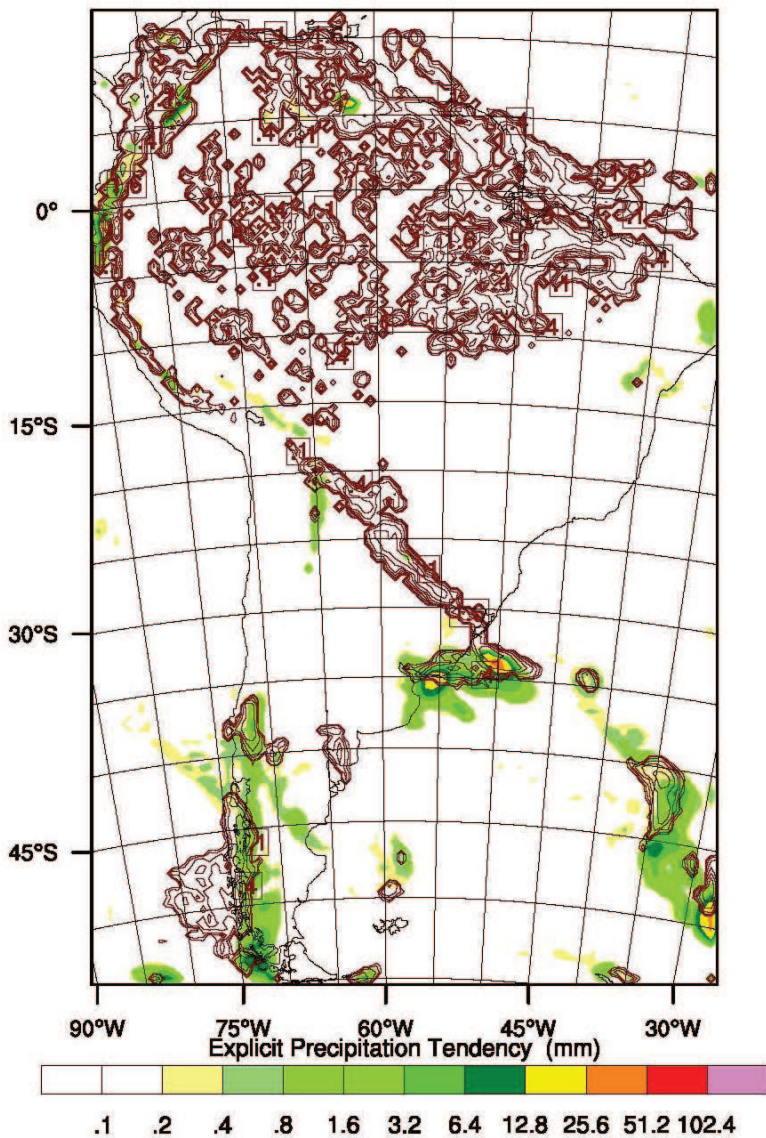


OUTPUT FROM WRF V3.2.1 MODEL  
 WE = 103 ; SN = 159 ; Levels = 28 ; Dis = 50.5km ; Phys Opt = 3 ; PBL Opt = 1 ; Cu Opt = 1



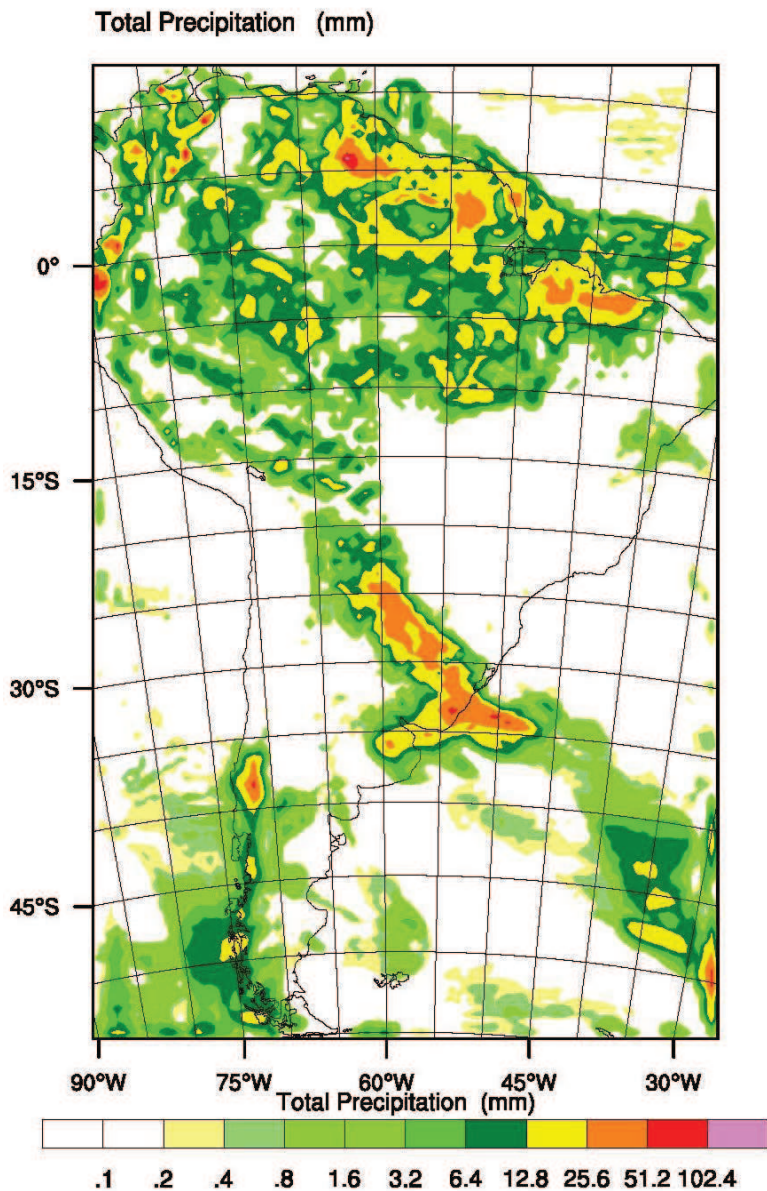
**REAL-TIME WRF**    Init: 2013-05-03\_12:00:00  
                          Valid: 2013-05-04\_00:07:12

Explicit Precipitation Tendency from 2013-05-03\_21:05:24 to 2013-05-04\_00:07:12 (mm)  
 Param Precipitation Tendency from 2013-05-03\_21:05:24 to 2013-05-04\_00:07:12 (mm)



OUTPUT FROM WRF V3.2.1 MODEL  
 WE = 103 ; SN = 159 ; Levels = 28 ; Dis = 50.5km ; Phys Opt = 3 ; PBL Opt = 1 ; Cu Opt = 1

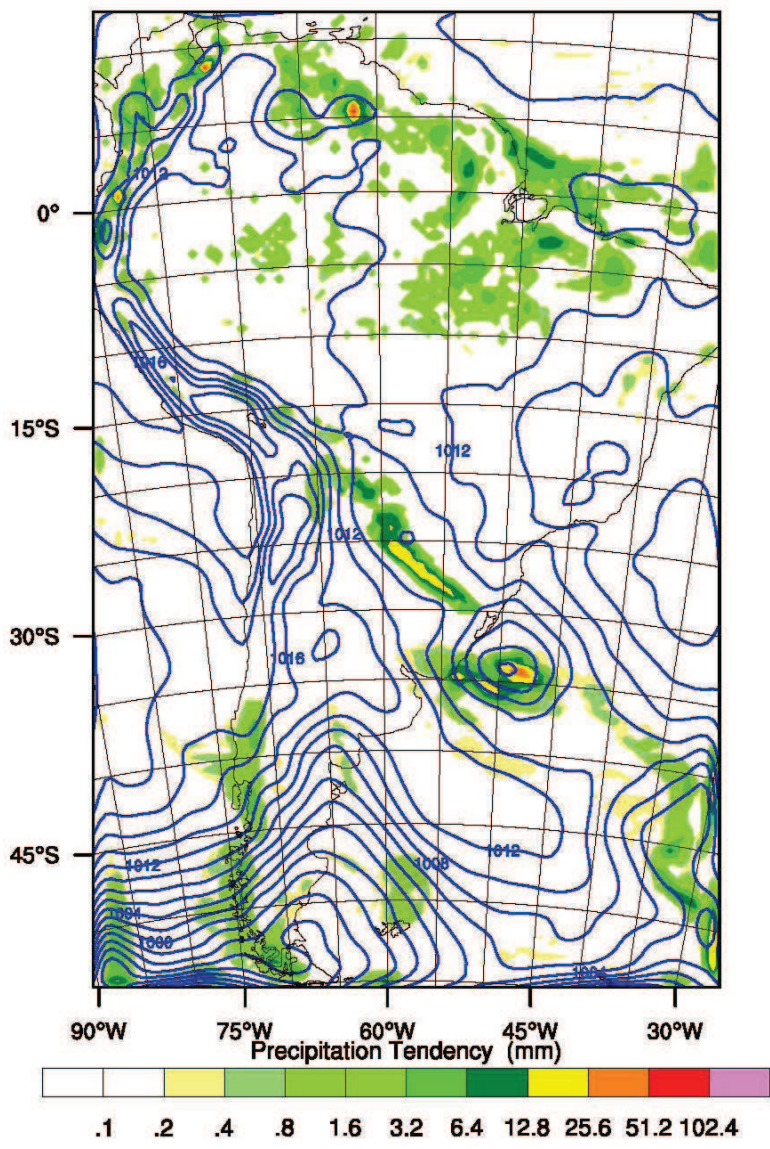
**REAL-TIME WRF**    Init: 2013-05-03\_12:00:00  
Valid: 2013-05-04\_03:09:00



OUTPUT FROM WRF V3.2.1 MODEL  
WE = 103 ; SN = 159 ; Levels = 28 ; Dis = 50.5km ; Phys Opt = 3 ; PBL Opt = 1 ; Cu Opt = 1

**REAL-TIME WRF**    Init: 2013-05-03\_12:00:00  
 Valid: 2013-05-04\_03:09:00

Precipitation Tendency from 2013-05-04\_00:07:12 to 2013-05-04\_03:09:00 (mm)  
 Sea Level Pressure (hPa)

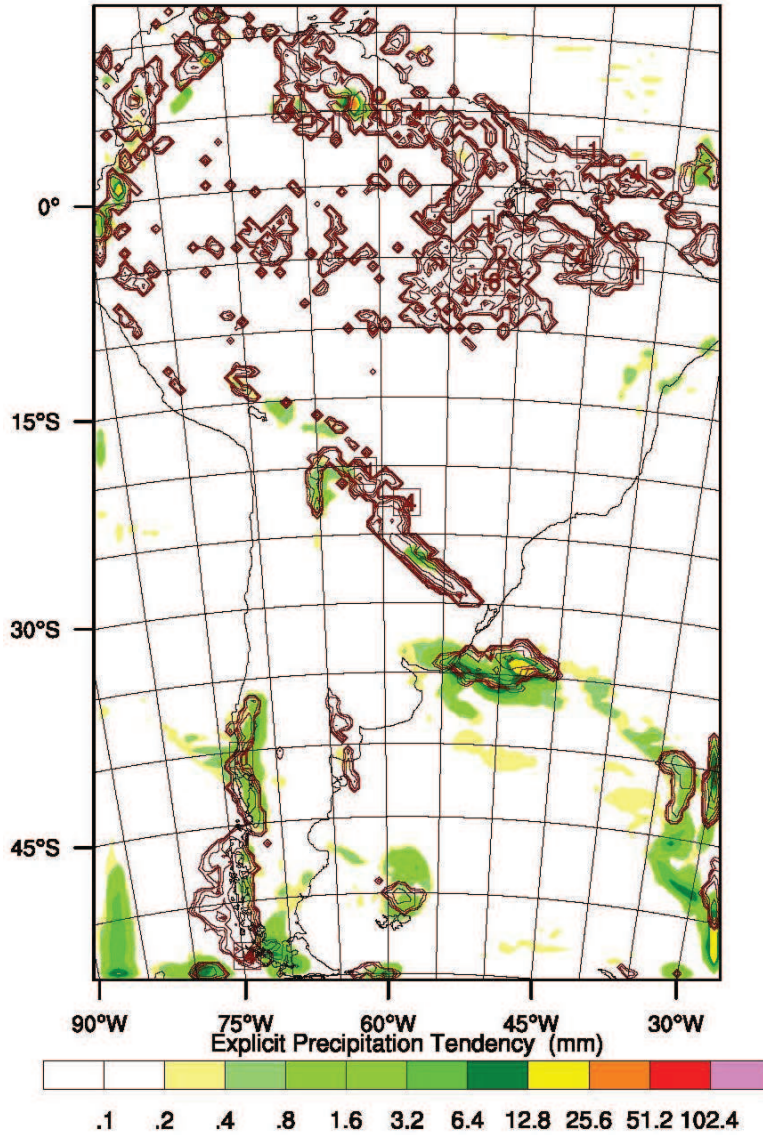


OUTPUT FROM WRF V3.2.1 MODEL  
 WE = 103 ; SN = 159 ; Levels = 28 ; Dis = 50.5km ; Phys Opt = 3 ; PBL Opt = 1 ; Cu Opt = 1



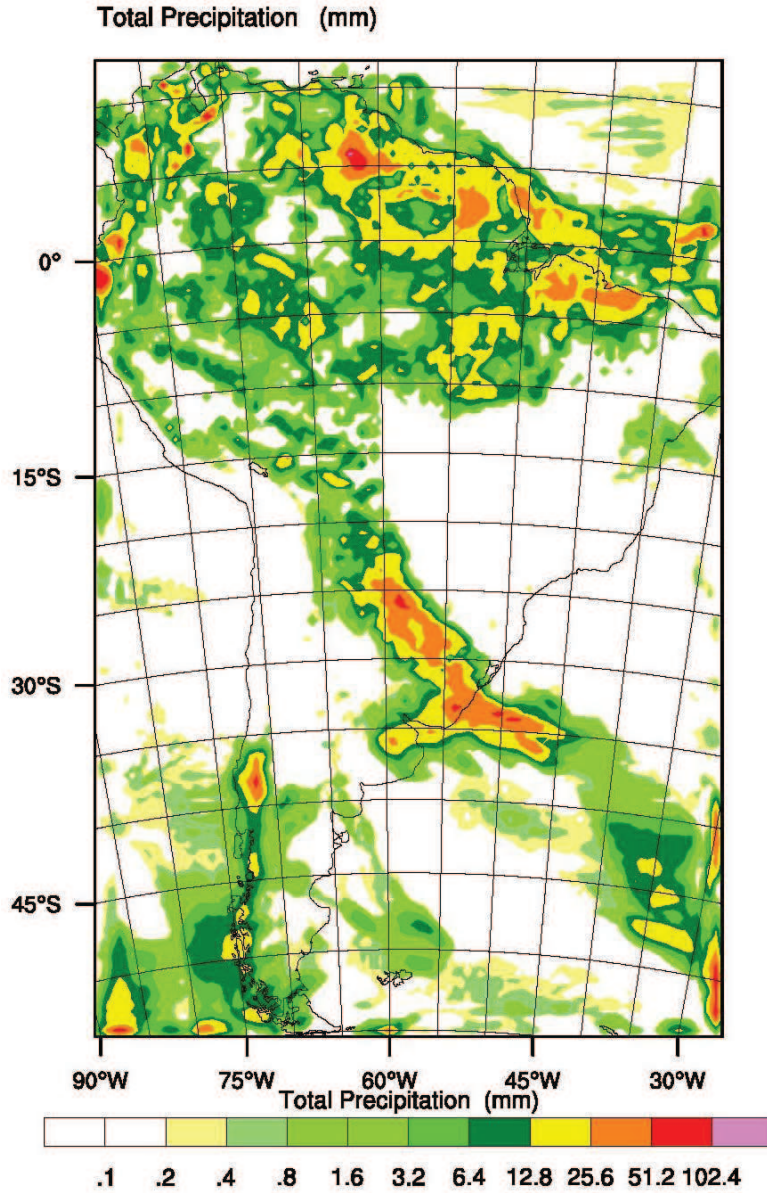
**REAL-TIME WRF**    Init: 2013-05-03\_12:00:00  
                          Valid: 2013-05-04\_03:09:00

Explicit Precipitation Tendency from 2013-05-04\_00:07:12 to 2013-05-04\_03:09:00 (mm)  
 Param Precipitation Tendency from 2013-05-04\_00:07:12 to 2013-05-04\_03:09:00 (mm)



OUTPUT FROM WRF V3.2.1 MODEL  
 WE = 103 ; SN = 159 ; Levels = 28 ; Dis = 50.5km ; Phys Opt = 3 ; PBL Opt = 1 ; Cu Opt = 1

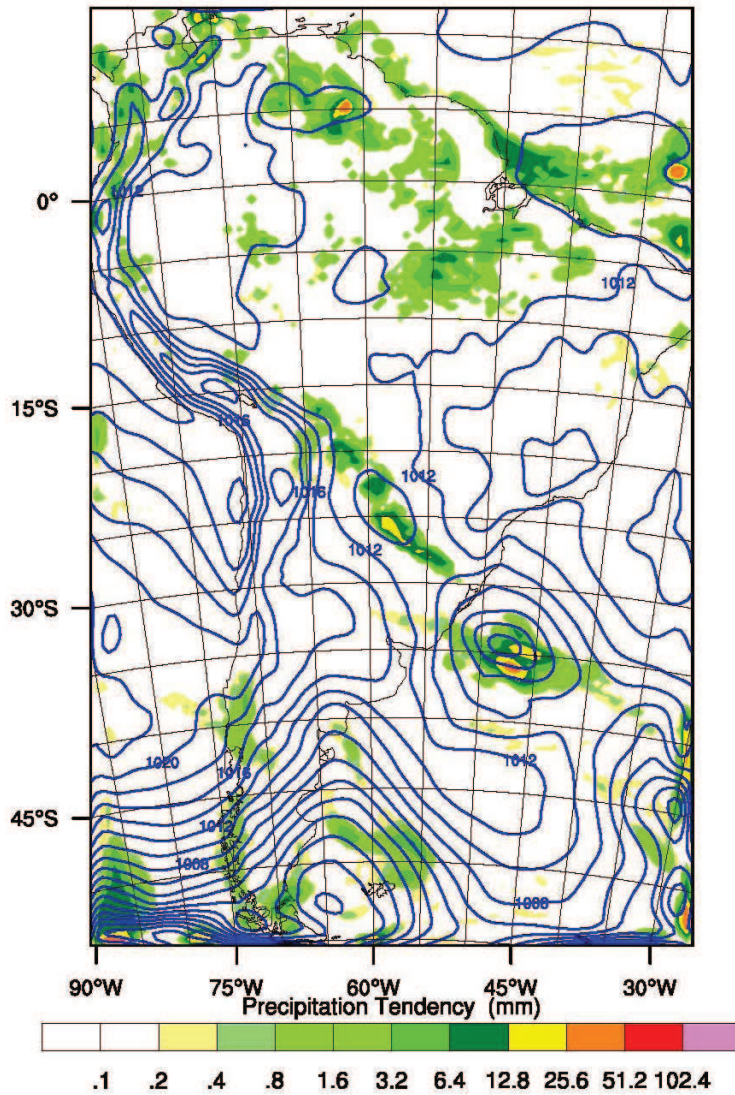
**REAL-TIME WRF**    Init: 2013-05-03\_12:00:00  
 Valid: 2013-05-04\_06:10:48



OUTPUT FROM WRF V3.2.1 MODEL  
 WE = 103 ; SN = 159 ; Levels = 28 ; Dis = 50.5km ; Phys Opt = 3 ; PBL Opt = 1 ; Cu Opt = 1

**REAL-TIME WRF**    Init: 2013-05-03\_12:00:00  
 Valid: 2013-05-04\_06:10:48

Precipitation Tendency from 2013-05-04\_03:09:00 to 2013-05-04\_06:10:48 (mm)  
 Sea Level Pressure (hPa)

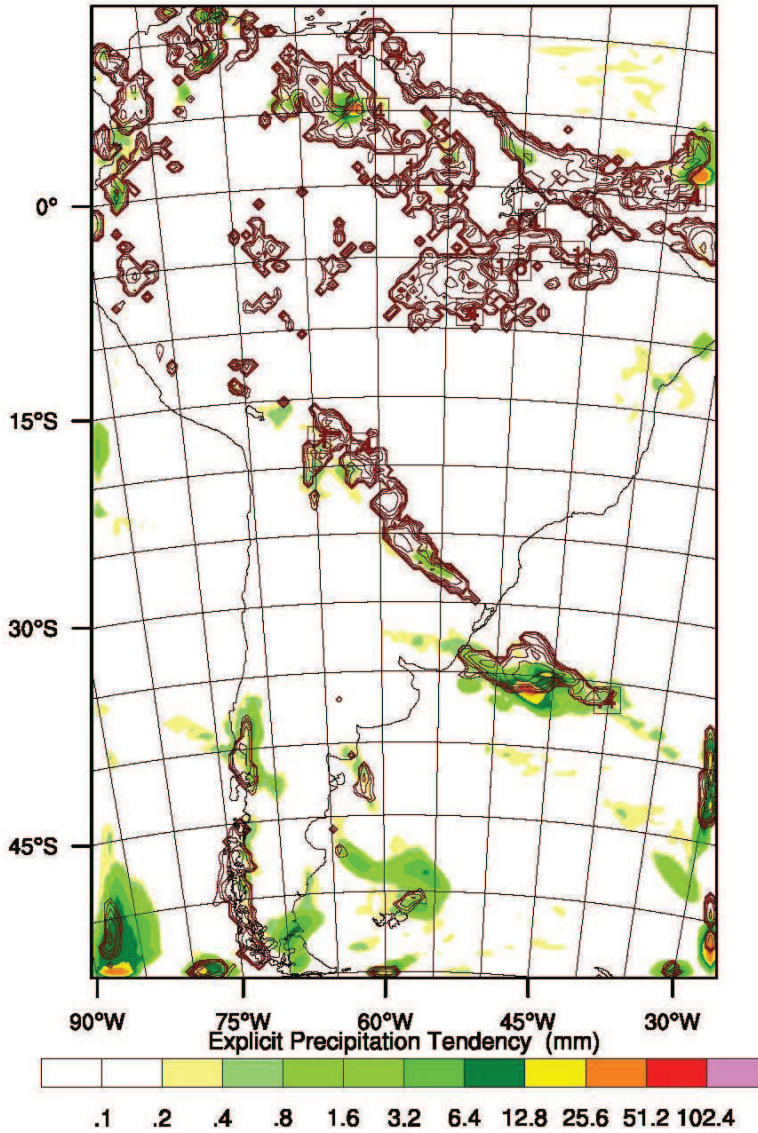


OUTPUT FROM WRF V3.2.1 MODEL  
 WE = 103 ; SN = 159 ; Levels = 28 ; Dis = 50.5km ; Phys Opt = 3 ; PBL Opt = 1 ; Cu Opt = 1



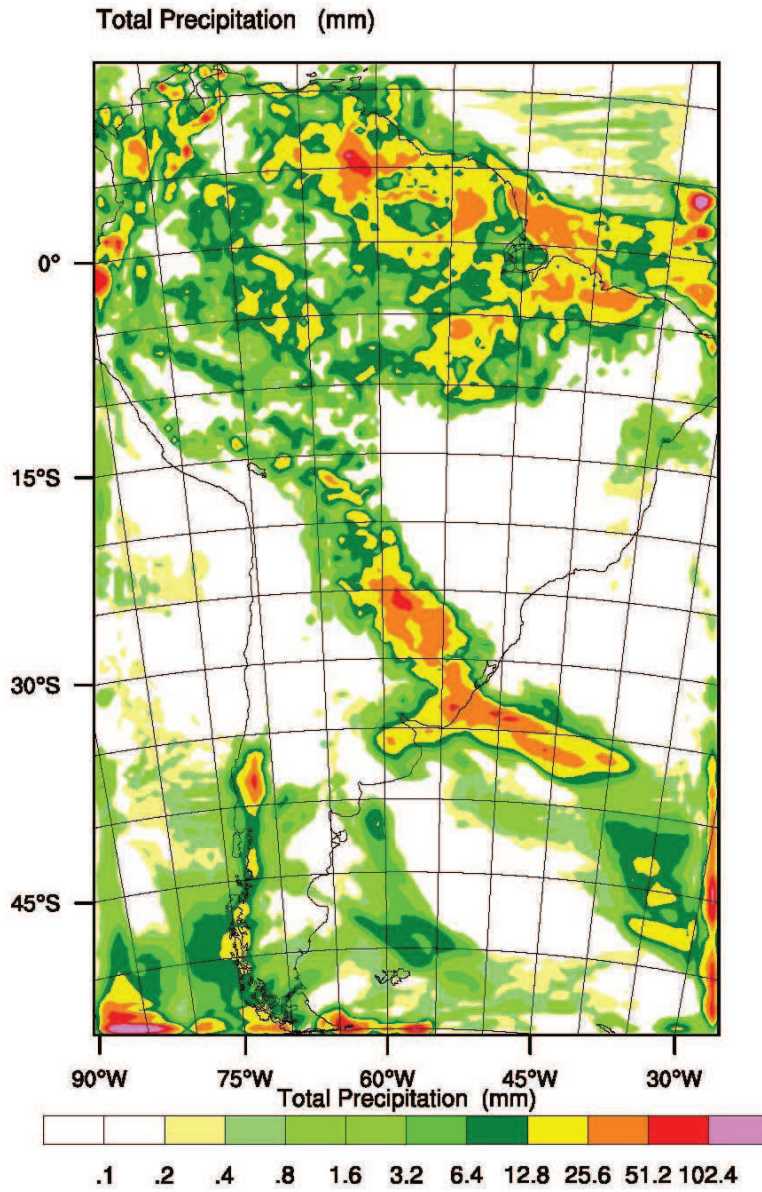
**REAL-TIME WRF**    Init: 2013-05-03\_12:00:00  
                          Valid: 2013-05-04\_06:10:48

Explicit Precipitation Tendency from 2013-05-04\_03:09:00 to 2013-05-04\_06:10:48 (mm)  
 Param Precipitation Tendency from 2013-05-04\_03:09:00 to 2013-05-04\_06:10:48 (mm)



OUTPUT FROM WRF V3.2.1 MODEL  
 WE = 103 ; SN = 159 ; Levels = 28 ; Dis = 50.5km ; Phys Opt = 3 ; PBL Opt = 1 ; Cu Opt = 1

**REAL-TIME WRF**    Init: 2013-05-03\_12:00:00  
 Valid: 2013-05-04\_12:14:24

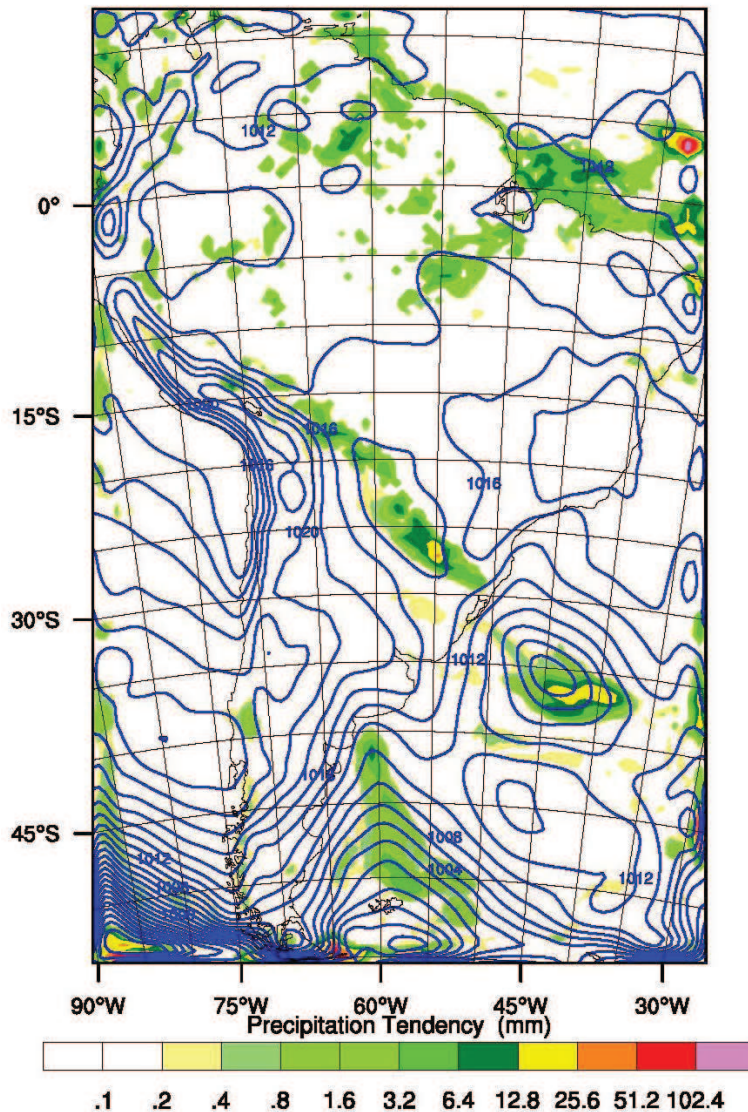


OUTPUT FROM WRF V3.2.1 MODEL  
 WE = 103 ; SN = 159 ; Levels = 28 ; Dis = 50.5km ; Phys Opt = 3 ; PBL Opt = 1 ; Cu Opt = 1



**REAL-TIME WRF**    Init: 2013-05-03\_12:00:00  
 Valid: 2013-05-04\_12:14:24

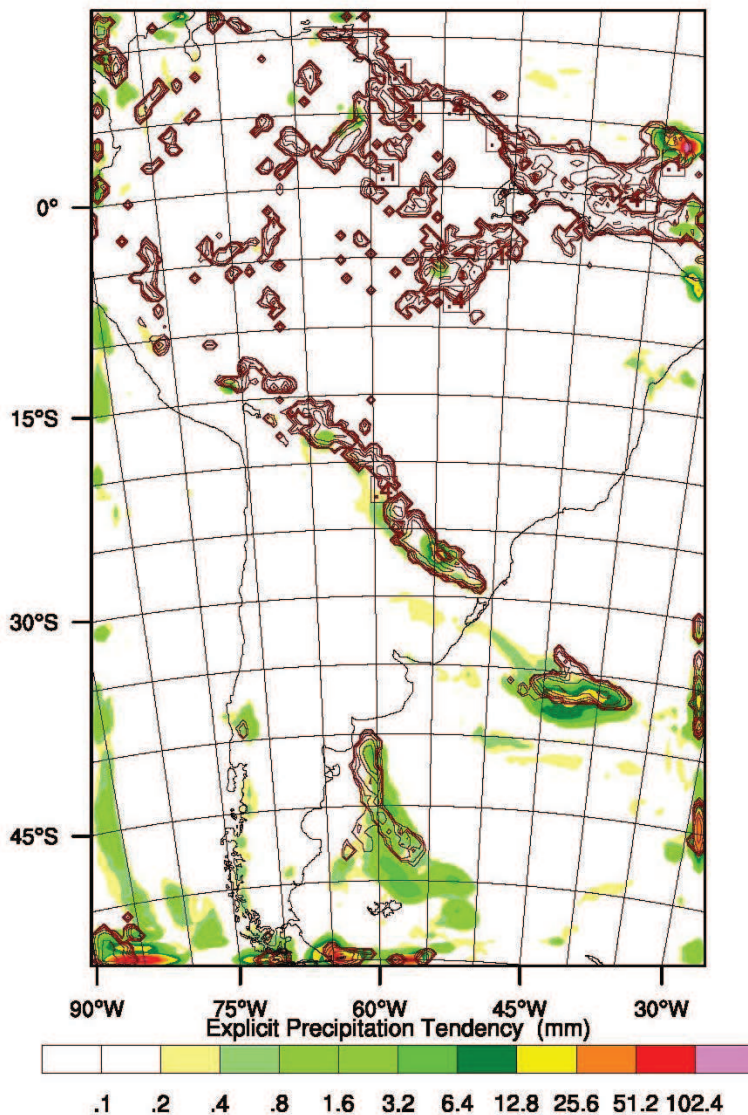
Precipitation Tendency from 2013-05-04\_09:12:36 to 2013-05-04\_12:14:24 (mm)  
 Sea Level Pressure (hPa)



OUTPUT FROM WRF V3.2.1 MODEL  
 WE = 103 ; SN = 159 ; Levels = 28 ; Dis = 50.5km ; Phys Opt = 3 ; PBL Opt = 1 ; Cu Opt = 1

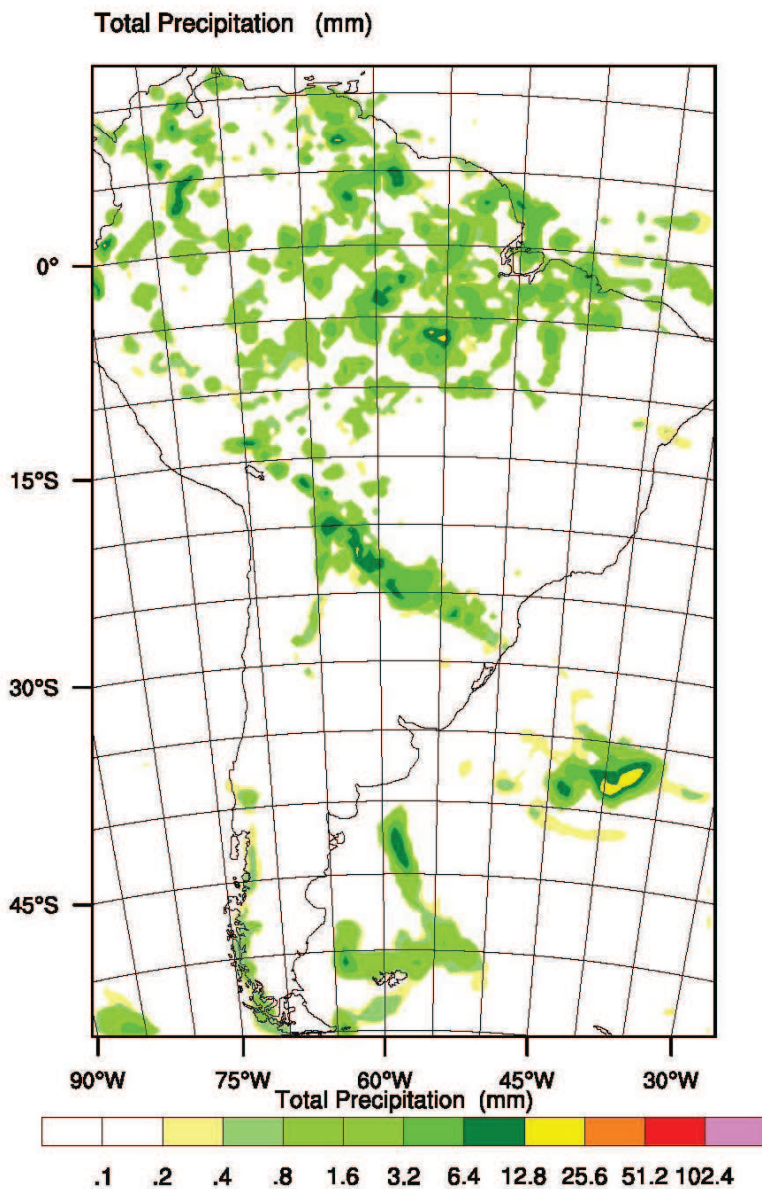
**REAL-TIME WRF**    Init: 2013-05-03\_12:00:00  
 Valid: 2013-05-04\_12:14:24

Explicit Precipitation Tendency from 2013-05-04\_09:12:36 to 2013-05-04\_12:14:24 (mm)  
 Param Precipitation Tendency from 2013-05-04\_09:12:36 to 2013-05-04\_12:14:24 (mm)



OUTPUT FROM WRF V3.2.1 MODEL  
 WE = 103 ; SN = 159 ; Levels = 28 ; Dis = 50.5km ; Phys Opt = 3 ; PBL Opt = 1 ; Cu Opt = 1

**REAL-TIME WRF**    Init: 2013-05-04\_12:00:00  
Valid: 2013-05-04\_15:01:48

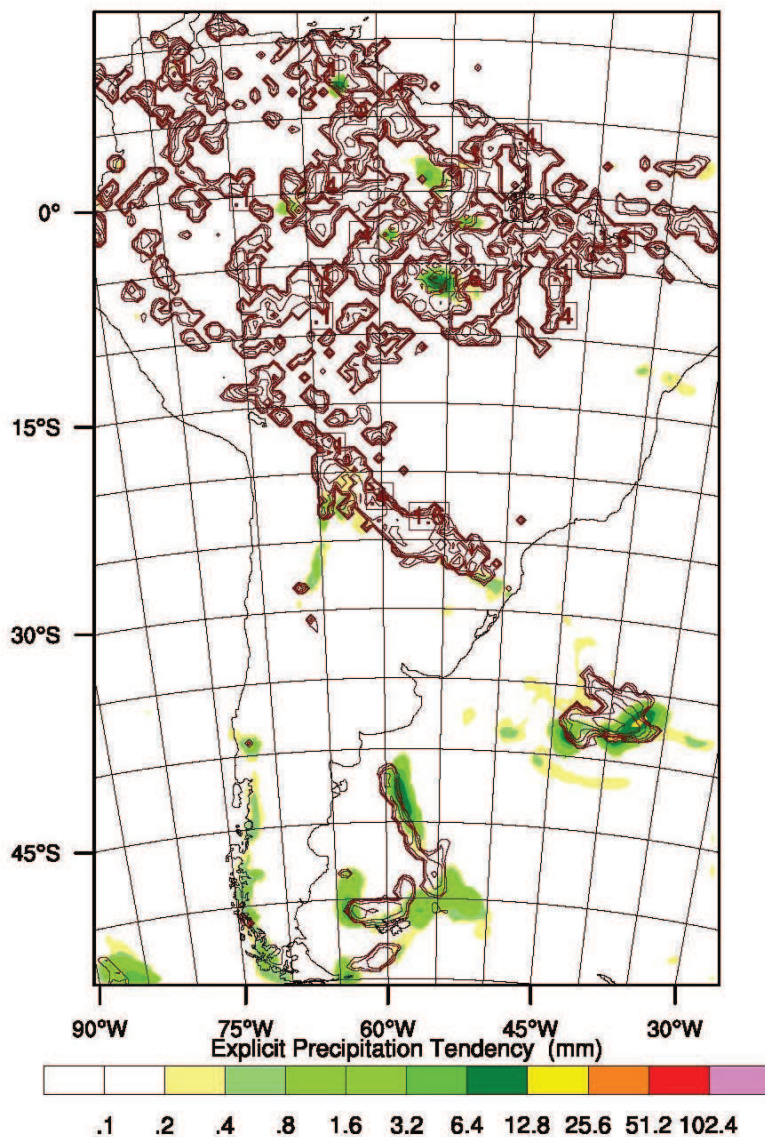


OUTPUT FROM WRF V3.2.1 MODEL  
WE = 103 ; SN = 159 ; Levels = 28 ; Dis = 50.5km ; Phys Opt = 3 ; PBL Opt = 1 ; Cu Opt = 1



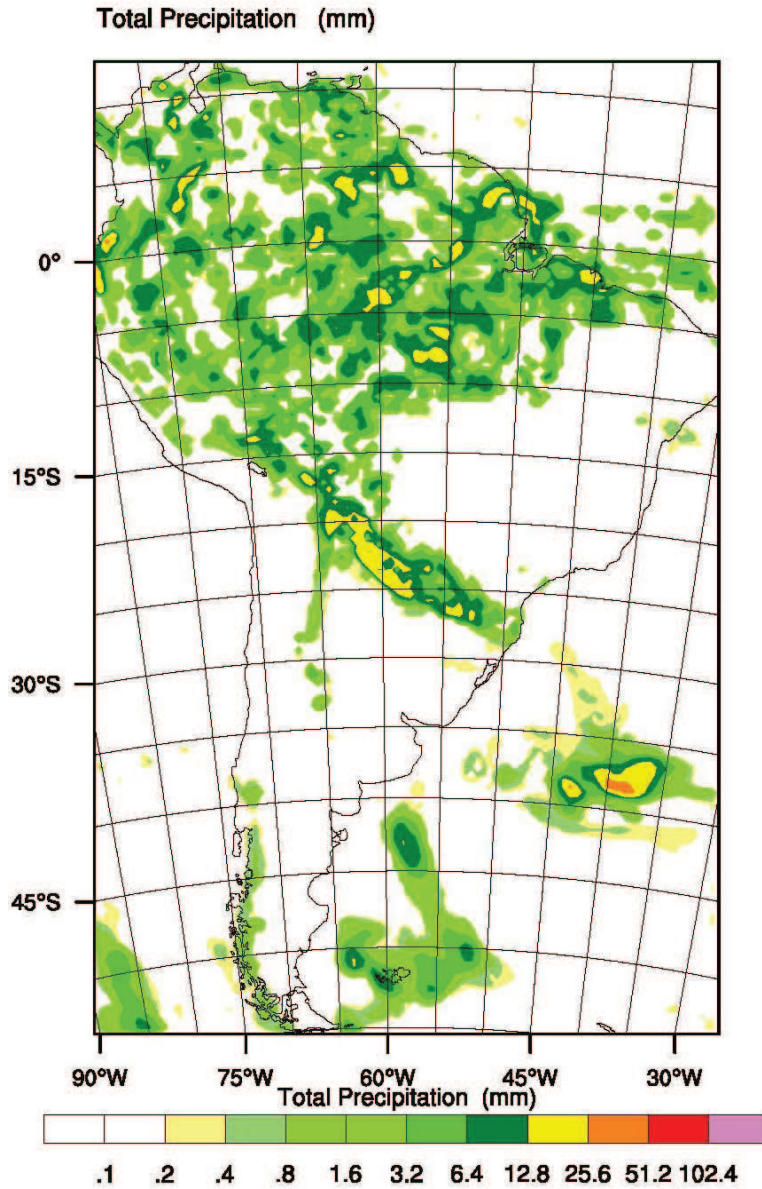
**REAL-TIME WRF**    Init: 2013-05-04\_12:00:00  
                              Valid: 2013-05-04\_15:01:48

Explicit Precipitation Tendency from 2013-05-04\_12:00:00 to 2013-05-04\_15:01:48 (mm)  
 Param Precipitation Tendency from 2013-05-04\_12:00:00 to 2013-05-04\_15:01:48 (mm)



OUTPUT FROM WRF V3.2.1 MODEL  
 WE = 103 ; SN = 159 ; Levels = 28 ; Dis = 50.5km ; Phys Opt = 3 ; PBL Opt = 1 ; Cu Opt = 1

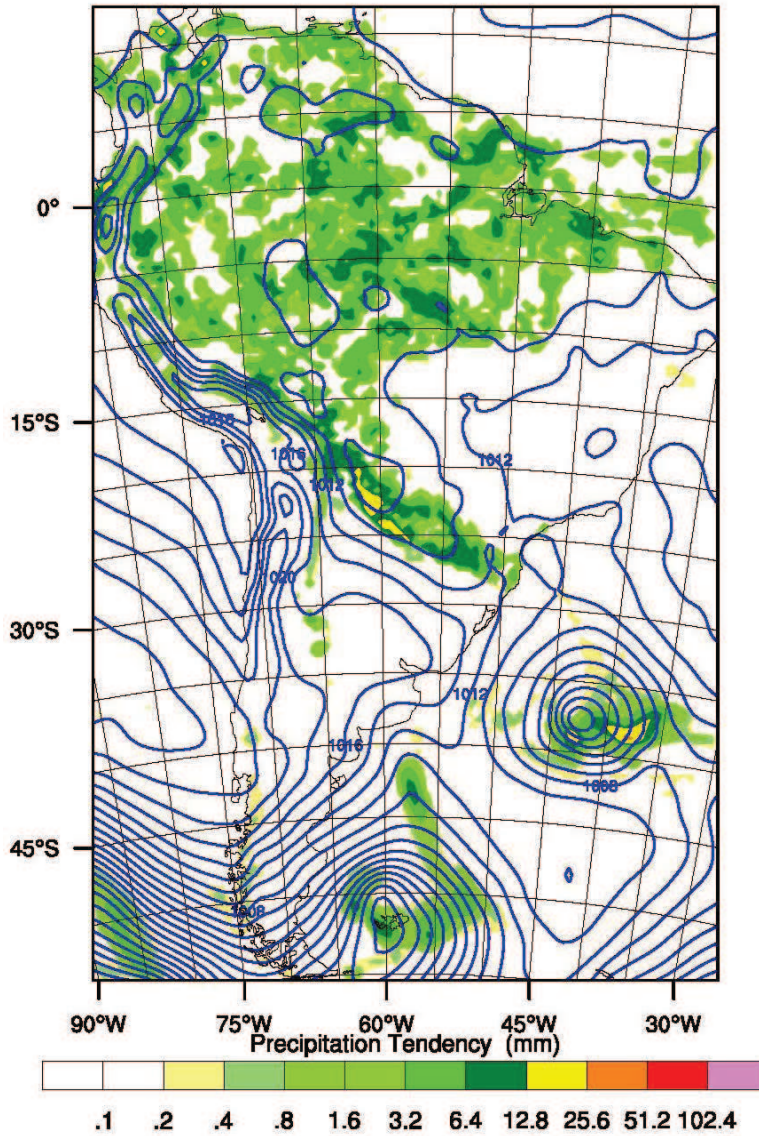
**REAL-TIME WRF**    Init: 2013-05-04\_12:00:00  
Valid: 2013-05-04\_18:03:36



OUTPUT FROM WRF V3.2.1 MODEL  
WE = 103 ; SN = 159 ; Levels = 28 ; Dis = 50.5km ; Phys Opt = 3 ; PBL Opt = 1 ; Cu Opt = 1

**REAL-TIME WRF**    Init: 2013-05-04\_12:00:00  
 Valid: 2013-05-04\_18:03:36

Precipitation Tendency from 2013-05-04\_15:01:48 to 2013-05-04\_18:03:36 (mm)  
 Sea Level Pressure (hPa)

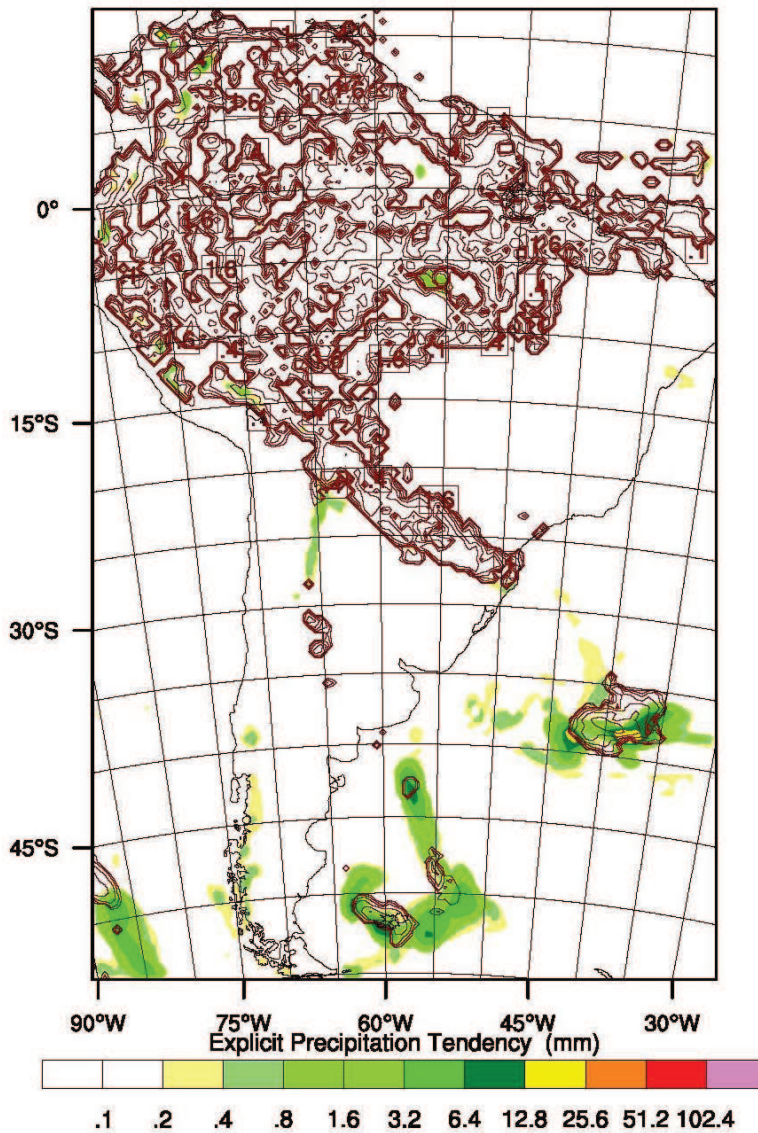


OUTPUT FROM WRF V3.2.1 MODEL  
 WE = 103 ; SN = 159 ; Levels = 28 ; Dis = 50.5km ; Phys Opt = 3 ; PBL Opt = 1 ; Cu Opt = 1



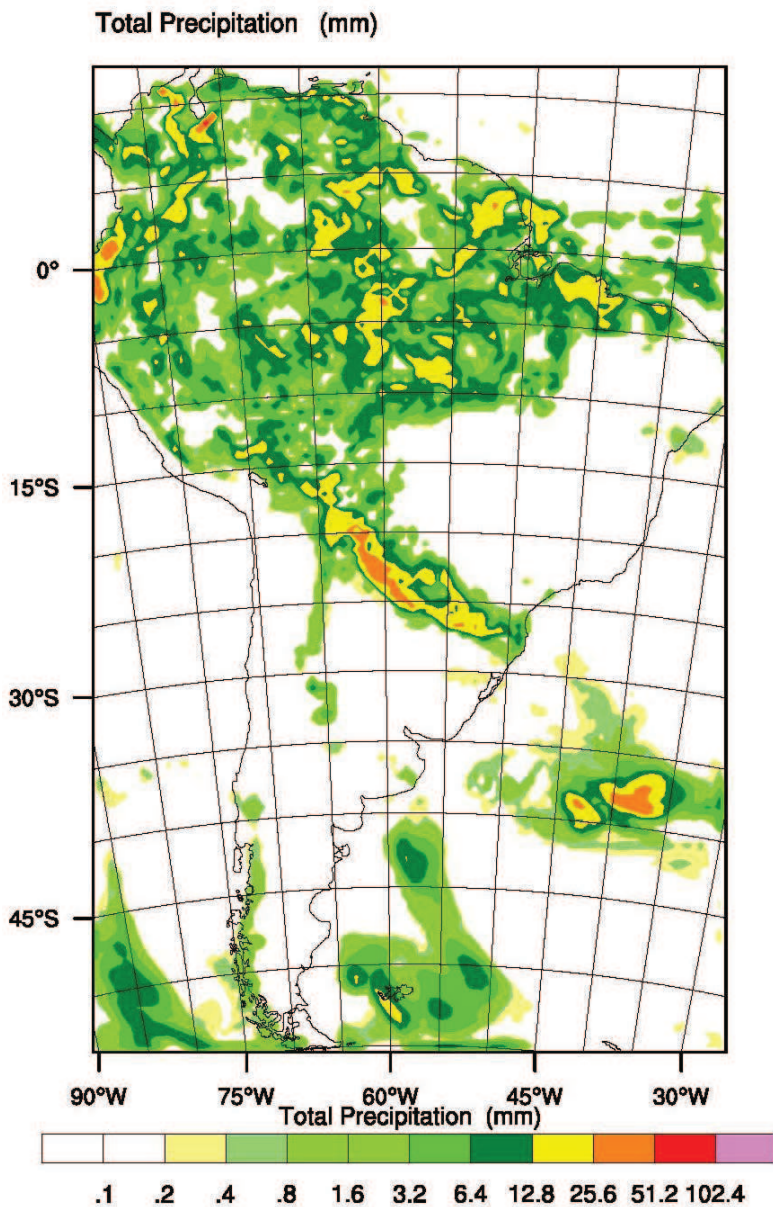
**REAL-TIME WRF**    Init: 2013-05-04\_12:00:00  
                          Valid: 2013-05-04\_18:03:36

Explicit Precipitation Tendency from 2013-05-04\_15:01:48 to 2013-05-04\_18:03:36 (mm)  
 Param Precipitation Tendency from 2013-05-04\_15:01:48 to 2013-05-04\_18:03:36 (mm)



OUTPUT FROM WRF V3.2.1 MODEL  
 WE = 103 ; SN = 159 ; Levels = 28 ; Dis = 50.5km ; Phys Opt = 3 ; PBL Opt = 1 ; Cu Opt = 1

**REAL-TIME WRF**    Init: 2013-05-04\_12:00:00  
 Valid: 2013-05-04\_21:05:24

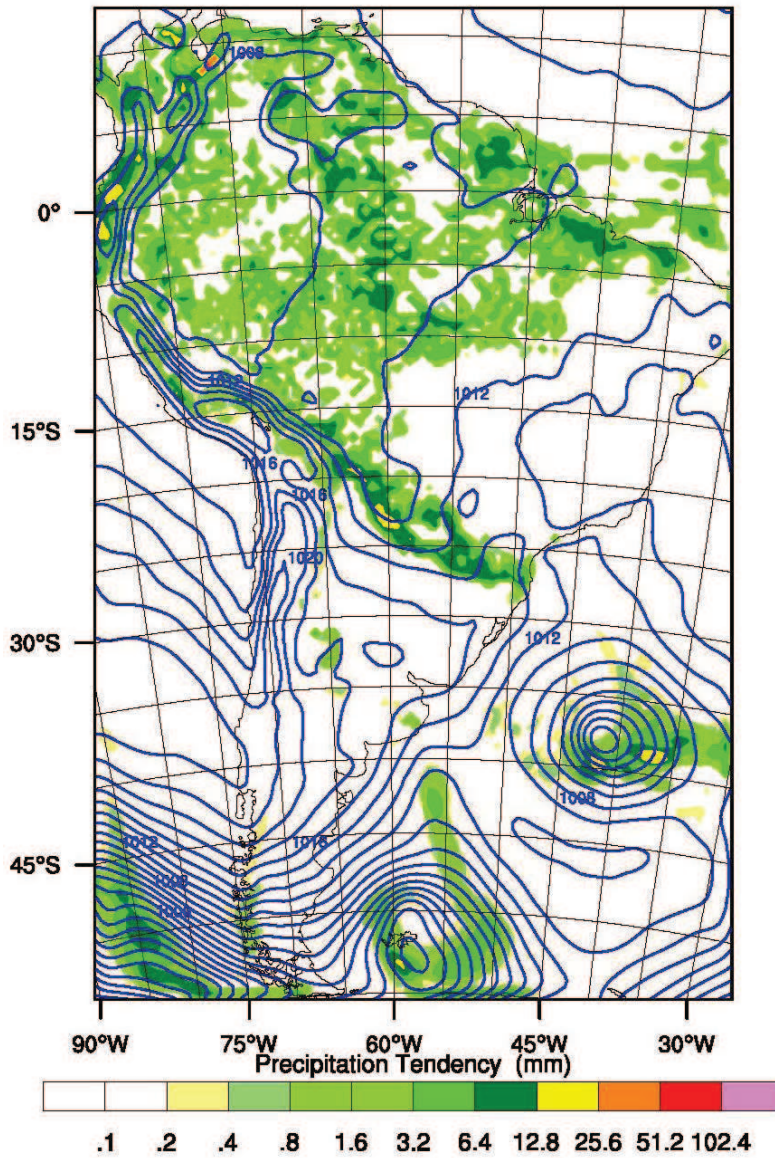


OUTPUT FROM WRF V3.2.1 MODEL  
 WE = 103 ; SN = 159 ; Levels = 28 ; Dis = 50.5km ; Phys Opt = 3 ; PBL Opt = 1 ; Cu Opt = 1



**REAL-TIME WRF**    Init: 2013-05-04\_12:00:00  
 Valid: 2013-05-04\_21:05:24

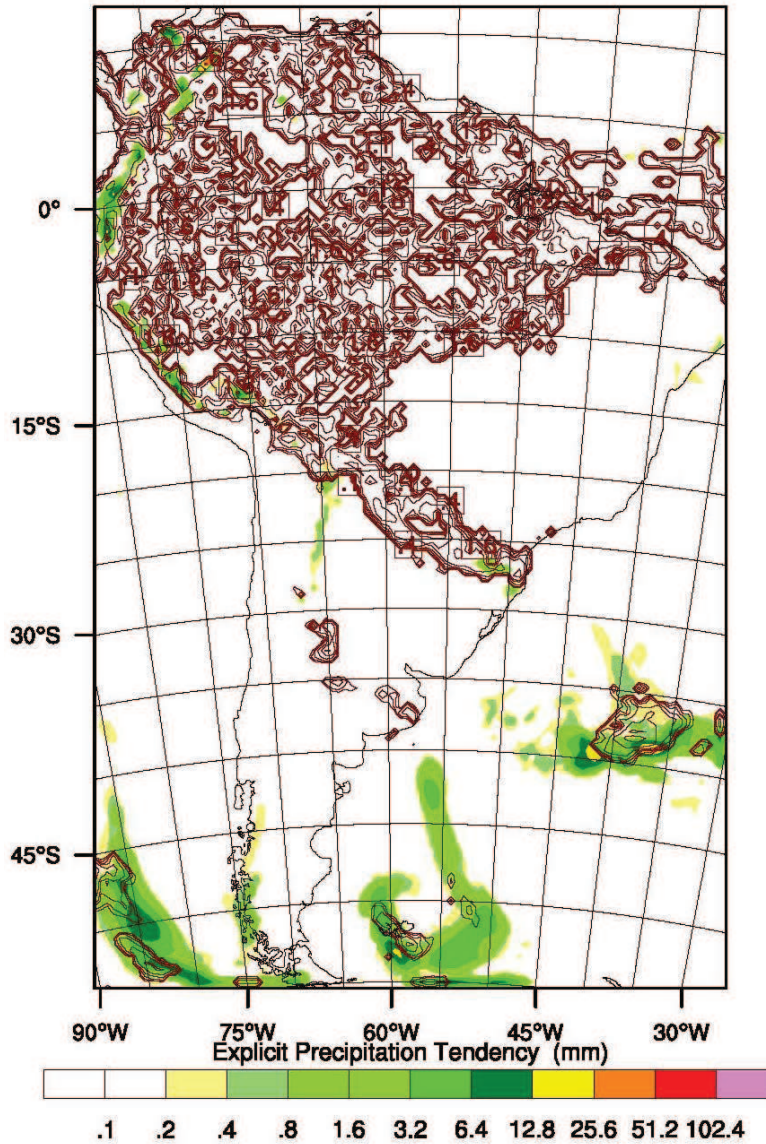
Precipitation Tendency from 2013-05-04\_18:03:36 to 2013-05-04\_21:05:24 (mm)  
 Sea Level Pressure (hPa)



OUTPUT FROM WRF V3.2.1 MODEL  
 WE = 103 ; SN = 159 ; Levels = 28 ; Dis = 50.5km ; Phys Opt = 3 ; PBL Opt = 1 ; Cu Opt = 1

**REAL-TIME WRF**    Init: 2013-05-04\_12:00:00  
Valid: 2013-05-04\_21:05:24

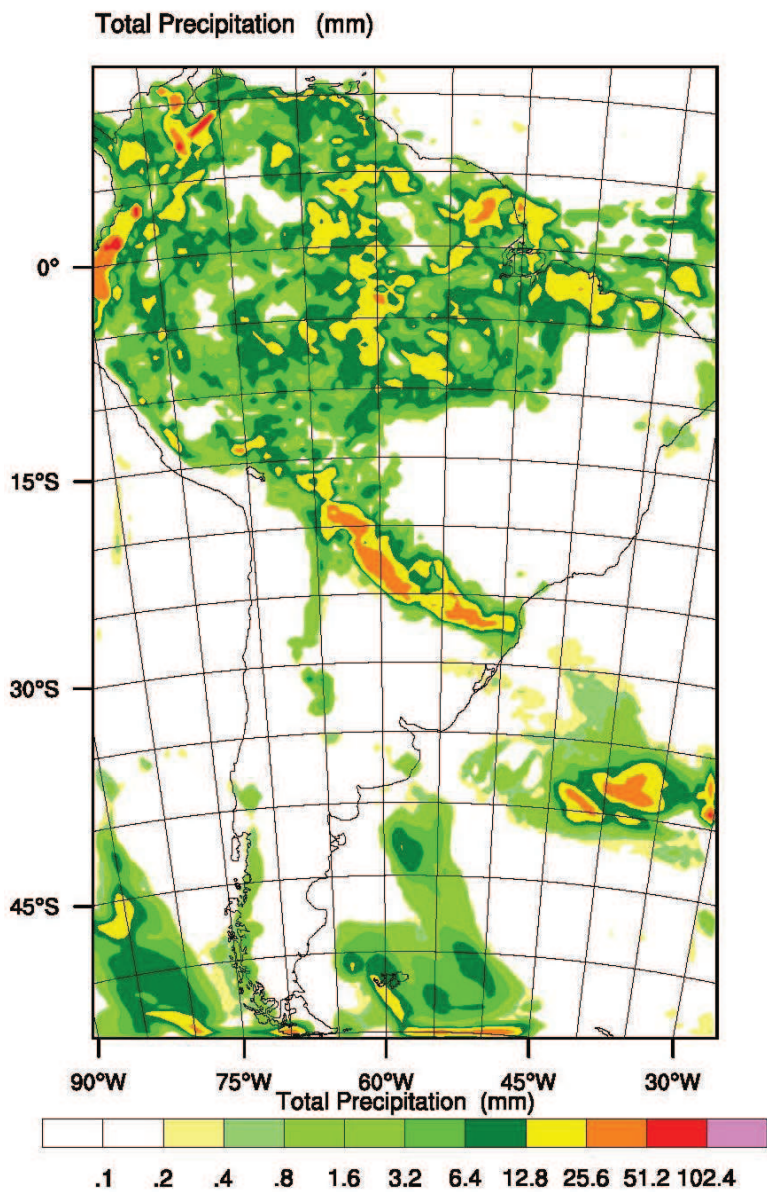
Explicit Precipitation Tendency from 2013-05-04\_18:03:36 to 2013-05-04\_21:05:24 (mm)  
Param Precipitation Tendency from 2013-05-04\_18:03:36 to 2013-05-04\_21:05:24 (mm)



OUTPUT FROM WRF V3.2.1 MODEL  
WE = 103 ; SN = 159 ; Levels = 28 ; Dis = 50.5km ; Phys Opt = 3 ; PBL Opt = 1 ; Cu Opt = 1



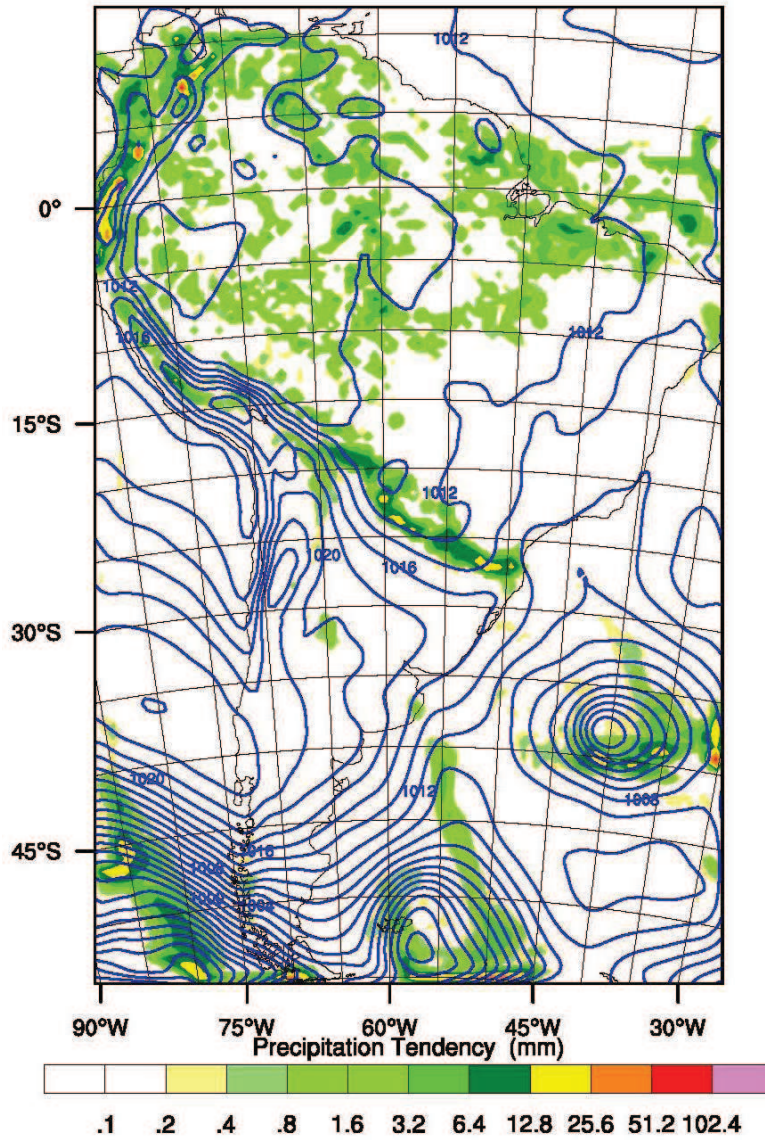
**REAL-TIME WRF**    Init: 2013-05-04\_12:00:00  
 Valid: 2013-05-05\_00:07:12



OUTPUT FROM WRF V3.2.1 MODEL  
 WE = 103 ; SN = 159 ; Levels = 28 ; Dis = 50.5km ; Phys Opt = 3 ; PBL Opt = 1 ; Cu Opt = 1

**REAL-TIME WRF**    Init: 2013-05-04\_12:00:00  
 Valid: 2013-05-05\_00:07:12

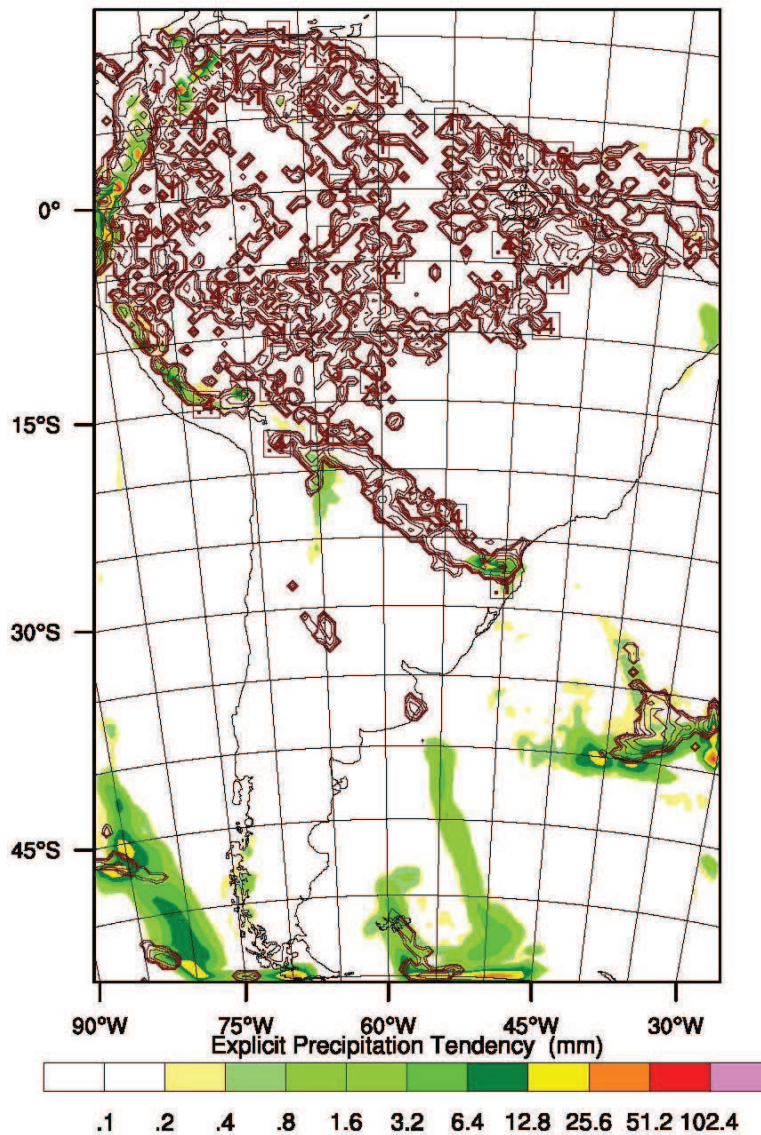
Precipitation Tendency from 2013-05-04\_21:05:24 to 2013-05-05\_00:07:12 (mm)  
 Sea Level Pressure (hPa)



OUTPUT FROM WRF V3.2.1 MODEL  
 WE = 103 ; SN = 159 ; Levels = 28 ; Dis = 50.5km ; Phys Opt = 3 ; PBL Opt = 1 ; Cu Opt = 1

**REAL-TIME WRF**    Init: 2013-05-04\_12:00:00  
Valid: 2013-05-05\_00:07:12

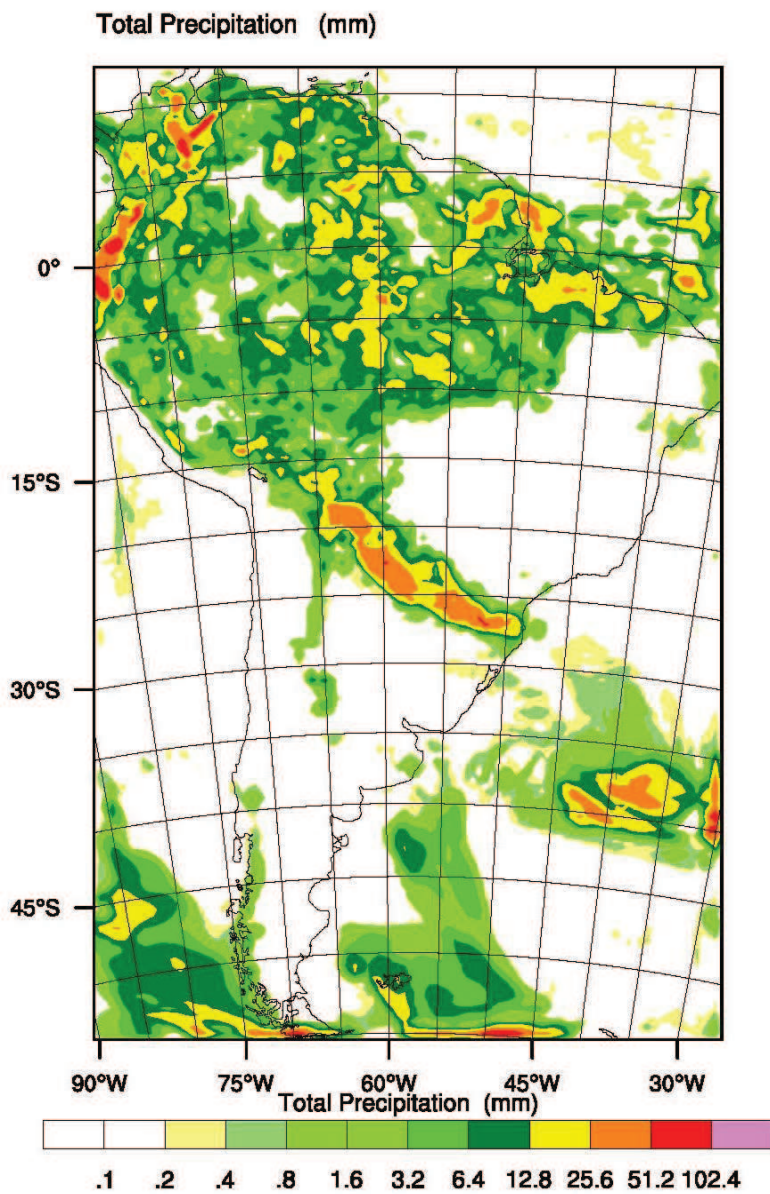
Explicit Precipitation Tendency from 2013-05-04\_21:05:24 to 2013-05-05\_00:07:12 (mm)  
Param Precipitation Tendency from 2013-05-04\_21:05:24 to 2013-05-05\_00:07:12 (mm)



OUTPUT FROM WRF V3.2.1 MODEL  
WE = 103 ; SN = 159 ; Levels = 28 ; Dis = 50.5km ; Phys Opt = 3 ; PBL Opt = 1 ; Cu Opt = 1



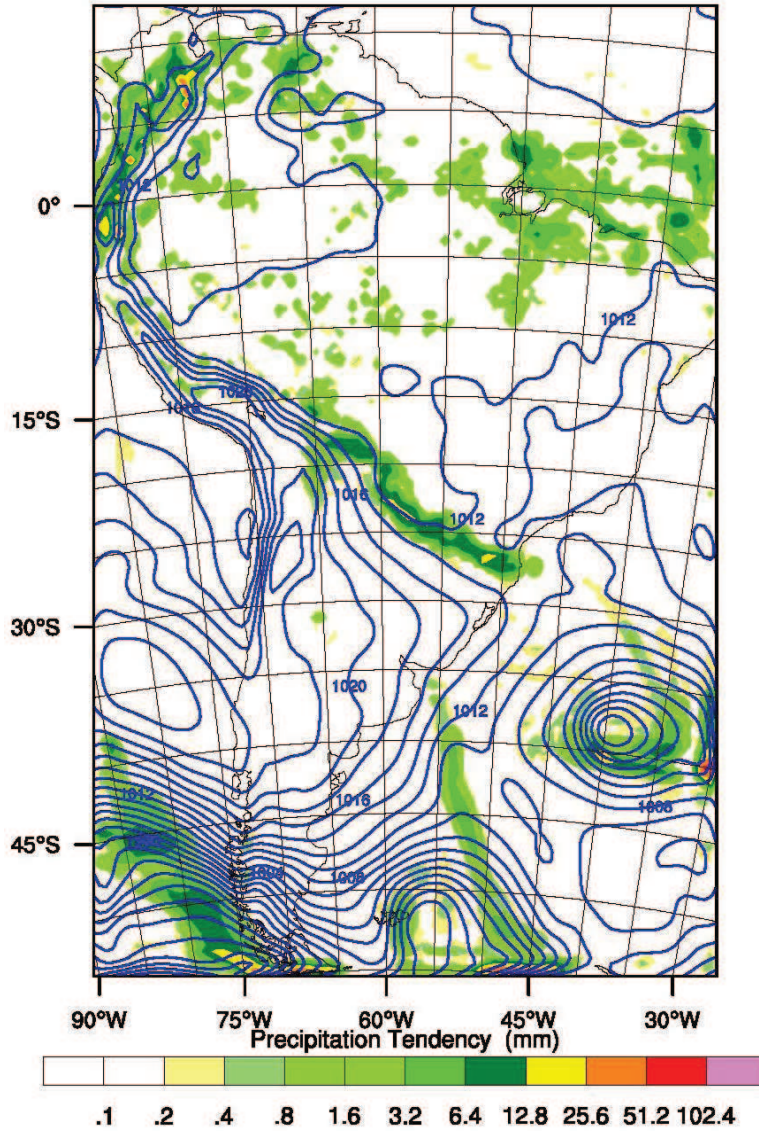
**REAL-TIME WRF**    Init: 2013-05-04\_12:00:00  
Valid: 2013-05-05\_03:09:00



OUTPUT FROM WRF V3.2.1 MODEL  
WE = 103 ; SN = 159 ; Levels = 28 ; Dis = 50.5km ; Phys Opt = 3 ; PBL Opt = 1 ; Cu Opt = 1

**REAL-TIME WRF**    Init: 2013-05-04\_12:00:00  
                              Valid: 2013-05-05\_03:09:00

Precipitation Tendency from 2013-05-05\_00:07:12 to 2013-05-05\_03:09:00 (mm)  
 Sea Level Pressure (hPa)

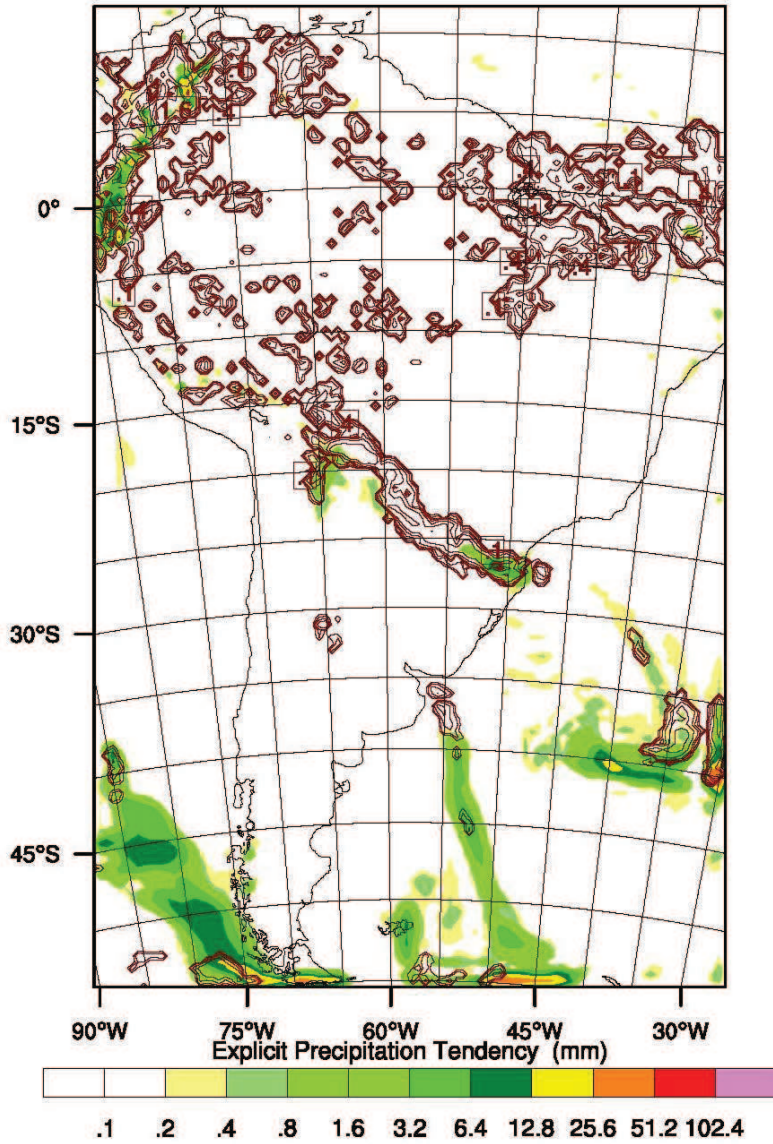


OUTPUT FROM WRF V3.2.1 MODEL  
 WE = 103 ; SN = 159 ; Levels = 28 ; Dis = 50.5km ; Phys Opt = 3 ; PBL Opt = 1 ; Cu Opt = 1



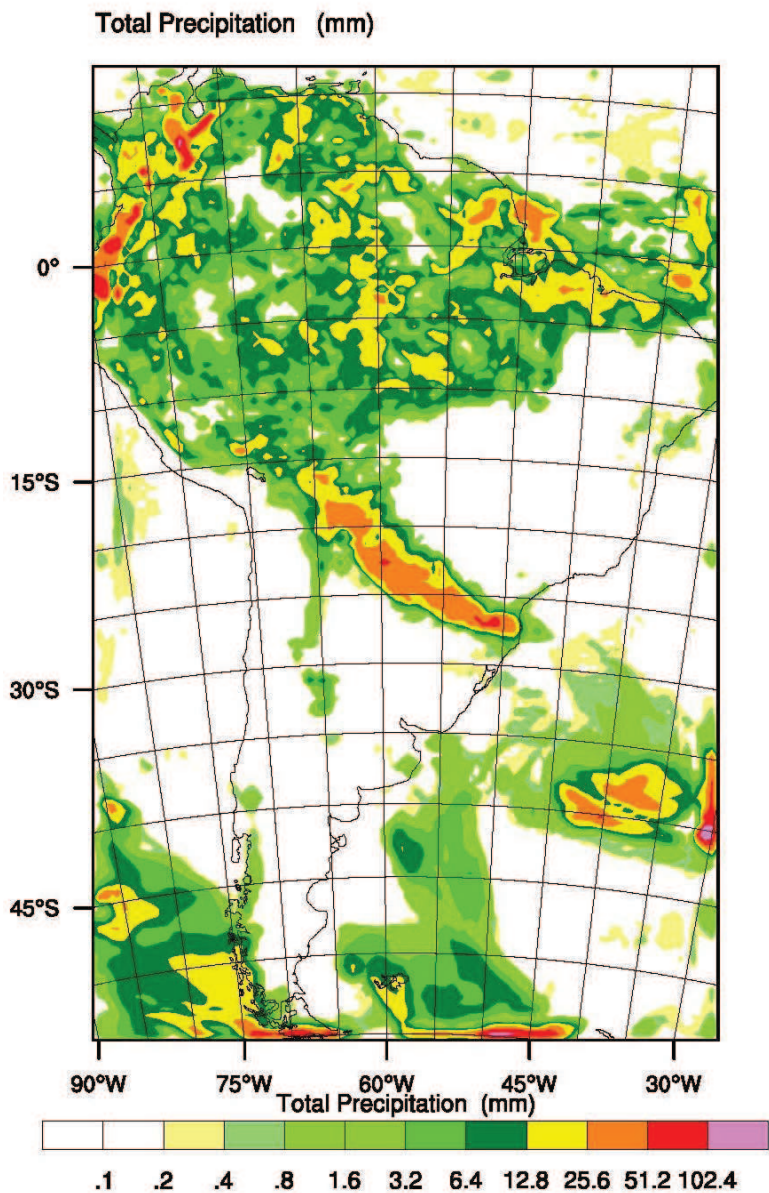
**REAL-TIME WRF**    Init: 2013-05-04\_12:00:00  
 Valid: 2013-05-05\_03:09:00

Explicit Precipitation Tendency from 2013-05-05\_00:07:12 to 2013-05-05\_03:09:00 (mm)  
 Param Precipitation Tendency from 2013-05-05\_00:07:12 to 2013-05-05\_03:09:00 (mm)



OUTPUT FROM WRF V3.2.1 MODEL  
 WE = 103 ; SN = 159 ; Levels = 28 ; Dis = 50.5km ; Phys Opt = 3 ; PBL Opt = 1 ; Cu Opt = 1

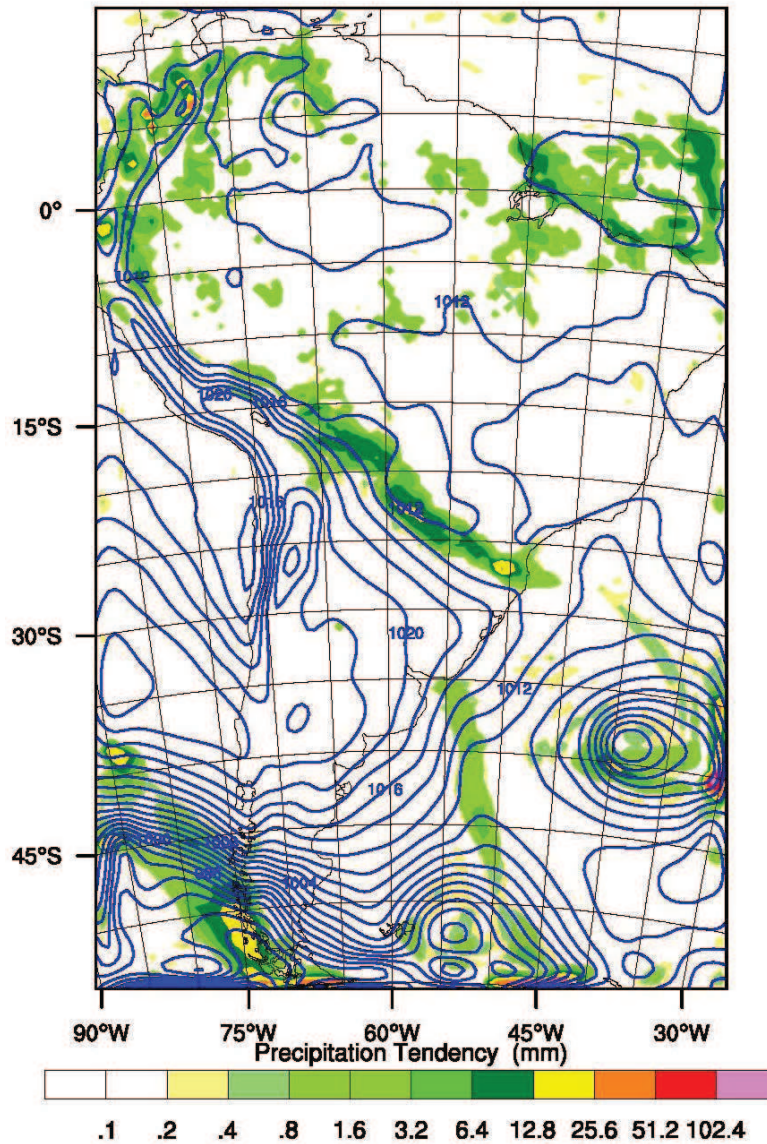
**REAL-TIME WRF**    Init: 2013-05-04\_12:00:00  
 Valid: 2013-05-05\_06:10:48



OUTPUT FROM WRF V3.2.1 MODEL  
 WE = 103 ; SN = 159 ; Levels = 28 ; Dis = 50.5km ; Phys Opt = 3 ; PBL Opt = 1 ; Cu Opt = 1

**REAL-TIME WRF**    Init: 2013-05-04\_12:00:00  
Valid: 2013-05-05\_06:10:48

Precipitation Tendency from 2013-05-05\_03:09:00 to 2013-05-05\_06:10:48 (mm)  
Sea Level Pressure (hPa)

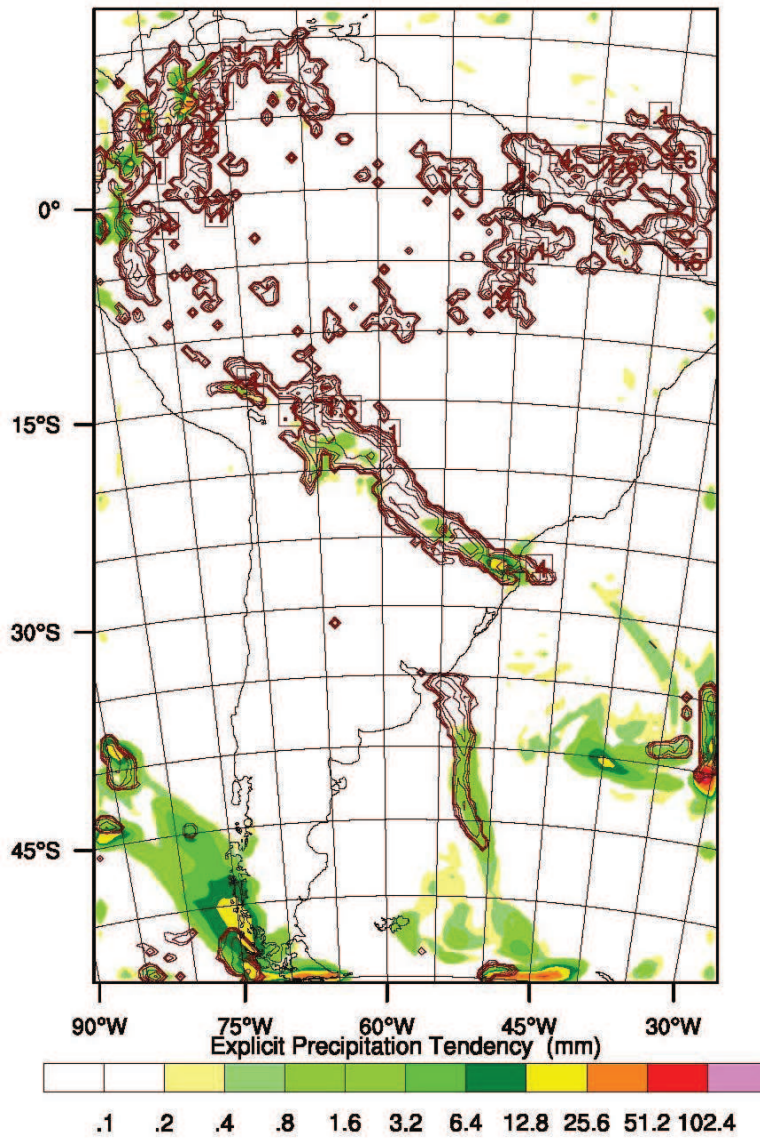


OUTPUT FROM WRF V3.2.1 MODEL  
WE = 103 ; SN = 159 ; Levels = 28 ; Dis = 50.5km ; Phys Opt = 3 ; PBL Opt = 1 ; Cu Opt = 1



**REAL-TIME WRF**    Init: 2013-05-04\_12:00:00  
Valid: 2013-05-05\_06:10:48

Explicit Precipitation Tendency from 2013-05-05\_03:09:00 to 2013-05-05\_06:10:48 (mm)  
Param Precipitation Tendency from 2013-05-05\_03:09:00 to 2013-05-05\_06:10:48 (mm)

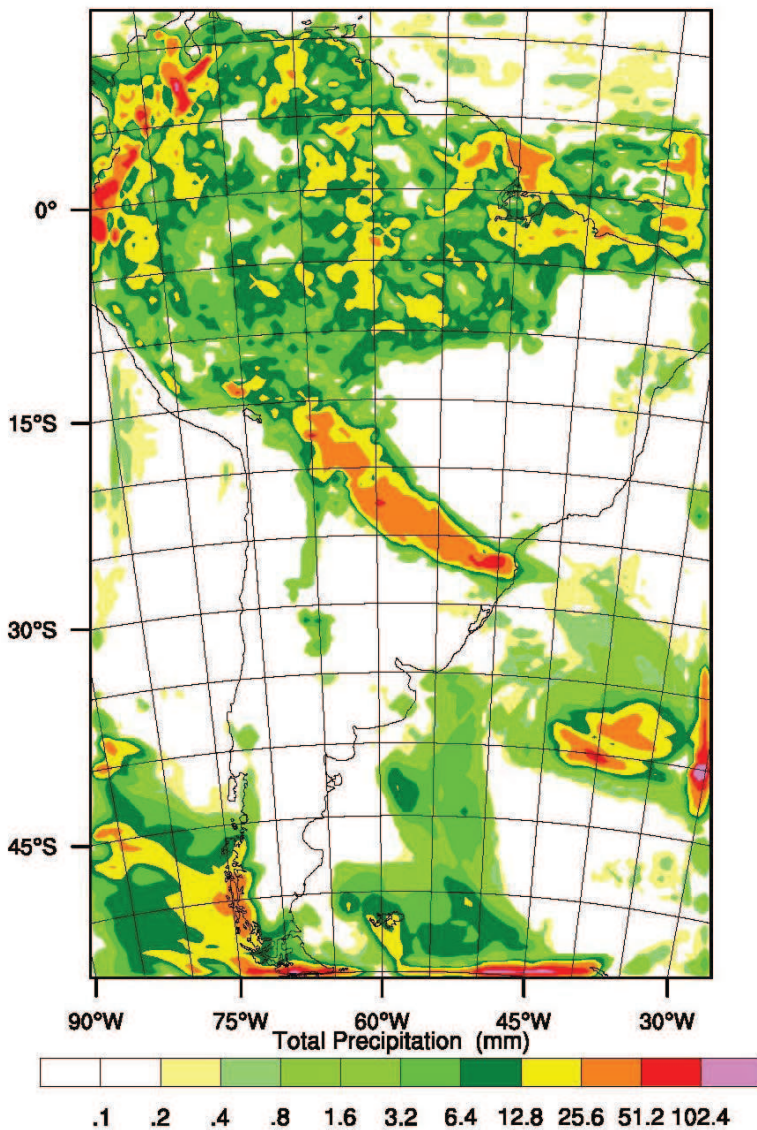


OUTPUT FROM WRF V3.2.1 MODEL

WE = 103 ; SN = 159 ; Levels = 28 ; Dis = 50.5km ; Phys Opt = 3 ; PBL Opt = 1 ; Cu Opt = 1

**REAL-TIME WRF**    Init: 2013-05-04\_12:00:00  
 Valid: 2013-05-05\_09:12:36

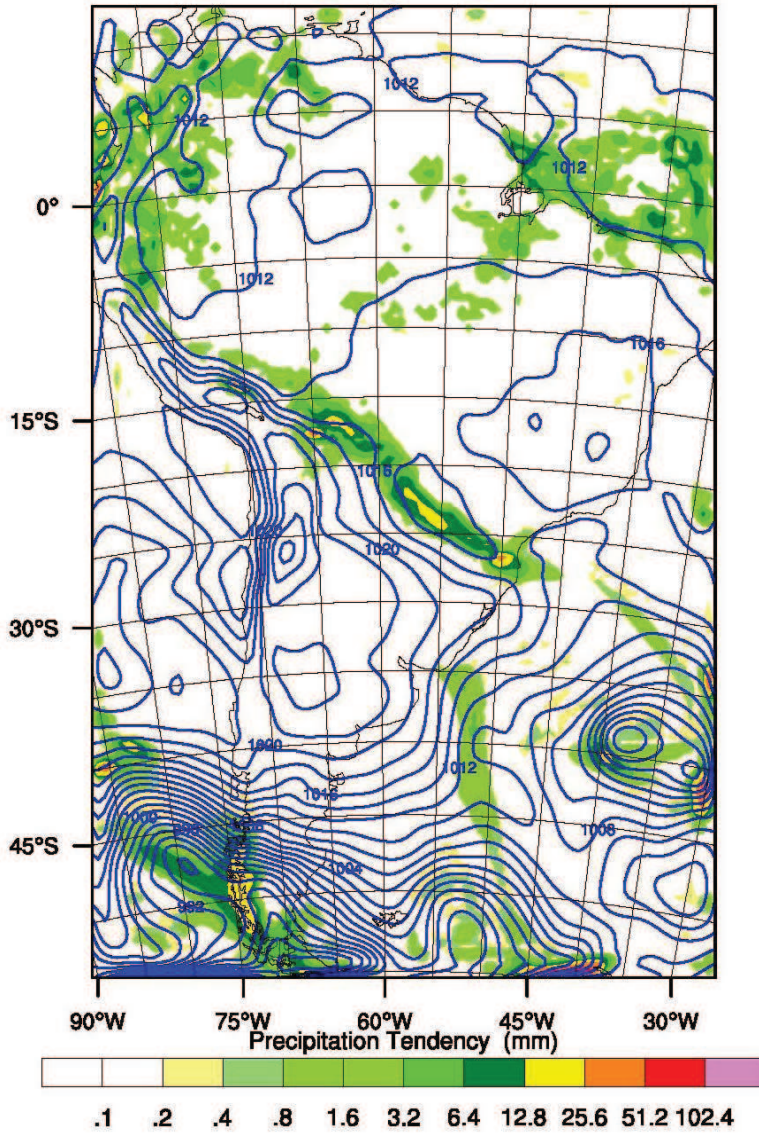
**Total Precipitation (mm)**



OUTPUT FROM WRF V3.2.1 MODEL  
 WE = 103 ; SN = 159 ; Levels = 28 ; Dis = 50.5km ; Phys Opt = 3 ; PBL Opt = 1 ; Cu Opt = 1

**REAL-TIME WRF**    Init: 2013-05-04\_12:00:00  
 Valid: 2013-05-05\_09:12:36

Precipitation Tendency from 2013-05-05\_06:10:48 to 2013-05-05\_09:12:36 (mm)  
 Sea Level Pressure (hPa)

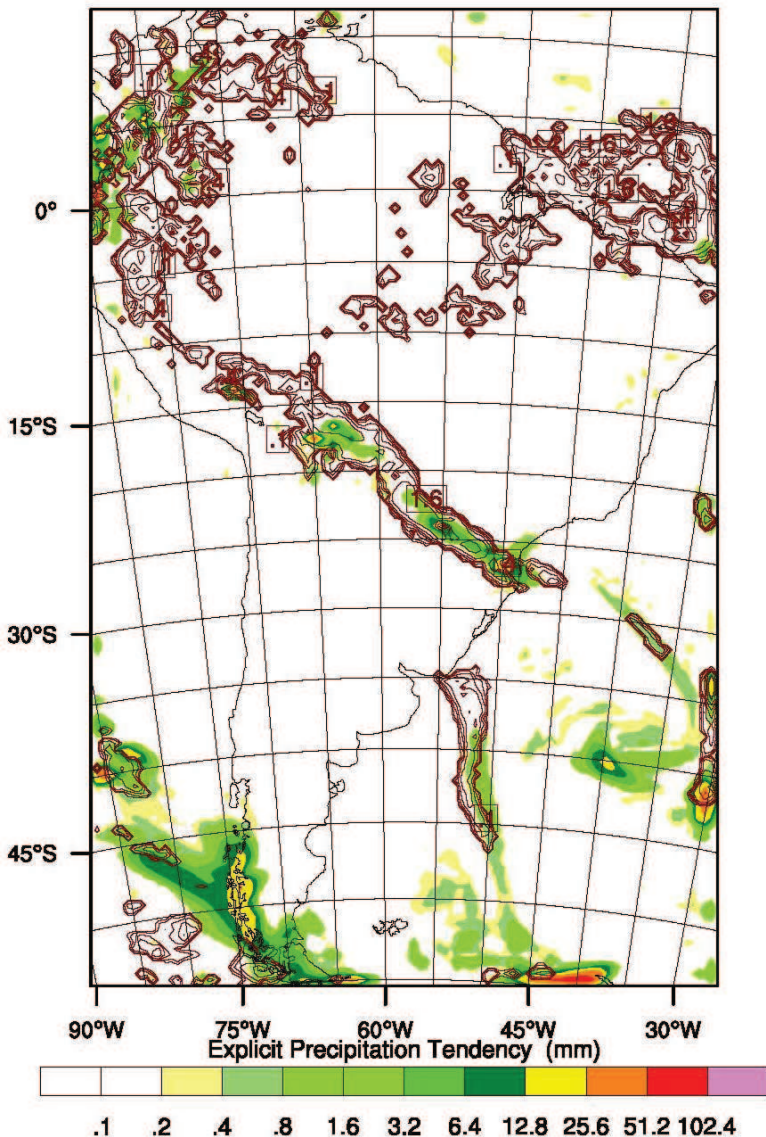


OUTPUT FROM WRF V3.2.1 MODEL  
 WE = 103 ; SN = 159 ; Levels = 28 ; Dis = 50.5km ; Phys Opt = 3 ; PBL Opt = 1 ; Cu Opt = 1



**REAL-TIME WRF**    Init: 2013-05-04\_12:00:00  
 Valid: 2013-05-05\_09:12:36

Explicit Precipitation Tendency from 2013-05-05\_06:10:48 to 2013-05-05\_09:12:36 (mm)  
 Param Precipitation Tendency from 2013-05-05\_06:10:48 to 2013-05-05\_09:12:36 (mm)



OUTPUT FROM WRF V3.2.1 MODEL  
 WE = 103 ; SN = 159 ; Levels = 28 ; Dis = 50.5km ; Phys Opt = 3 ; PBL Opt = 1 ; Cu Opt = 1







