



Grupo de Análisis de situaciones  
Meteorológicas Adversas (GAMA)



Universidad de Barcelona  
Departamento de Astronomía y Meteorología

Contribución del grupo 05 a la Red Ibérica para la investigación y desarrollo de aplicaciones en base al modelo atmosférico MM5 (REN2002-10018-E/CLI).

# Manual de instalación del MM5 usando los compiladores FORTRAN de INTEL (IFC).

Barrera, A., M.A. Prat y  
M.C. Llasat

Barcelona, mayo de 2004

DAM/250999-14/0405



# ÍNDICE

<b>1. INTRODUCCIÓN</b>	<b>2</b>
1.1 Requerimientos de hardware y software	3
1.1.1 LINUX	3
1.1.2 Compiladores FORTRAN de INTEL (IFC)	3
1.1.3 NCAR Graphics	4
1.1.4 VIS5D+	4
1.1.5 MM5	5
<b>2. INSTALACIÓN DE LOS REQUERIMIENTOS</b>	<b>7</b>
2.1 Sistema Operativo LINUX	7
2.2 Esquema general	19
2.3 Compiladores FORTRAN	21
2.4 NCAR Graphics	22
2.5 VIS5D+	24
2.5.1 VIS5D+ (ver-1.1)	24
• Uso de funciones externas con Vis5D+ (userfuncs )	26
2.5.2 VIS5D+ (ver-1.2.1)	28
<b>3. INSTALACIÓN DE LOS MÓDULOS DEL MM5</b>	<b>31</b>
3.1 TERRAIN	31
3.2 REGRID	37
3.2.1 Ejecución de pregrid	41
3.2.2 Ejecución de regridder	42
3.3 RAWINS	42
3.4 FETCH	45
3.5 LITTLE_R	49
3.6 INTERPF	53
3.7 MM5	55
3.8 INTERPB	58
3.9 NESTDOWN	61
3.10 PAQUETES GRÁFICOS	72
3.10.1 Conversión al VIS5D+	73
• MM5p_to_v5d	73
• REGRID_to_v5d	76
• TOVIS5D	79
3.10.2 Conversión al SURFER	81
3.10.3 Conversión al GrADS (MM5toGrADS)	83



## 1. INTRODUCCIÓN

Las presentes notas recopilan los pasos a seguir para poder compilar y ejecutar el modelo MM5 con el compilador gratuito FORTRAN de INTEL (versión 7.1) sobre un PC Linux Red Hat 9. Para versiones anteriores de Red Hat (6, 7 u 8) y del compilador (versión 6), véase la siguiente dirección:

- <http://acd.ufrj.br/~ricardom/mm5>.

Los pasos explicados en estas notas se han seguido con éxito por el Grupo de Análisis de Situaciones Meteorológicas Adversas (GAMA) del Departamento de Astronomía y Meteorología de la Facultad de Física de la Universidad de Barcelona (España). Este grupo forma parte de la Red Ibérica para la investigación y desarrollo de aplicaciones en base al modelo atmosférico MM5 (REN2002-10018-E/CLI) (Grupo 05).

Nuestro agradecimiento al Sr. Ricardo Marcelo da Silva (Universidade Federal do Rio de Janeiro, Brasil) por su estimable ayuda, sin la cual no hubiera sido posible gran parte del contenido de las presentes notas.

También queremos dar las gracias al Dr. Luis Cesáreo Cana Cascallar (Universidad de las Palmas de Gran Canaria, España), al Sr. Joan Miquel Torres Mari y al Dr. Romualdo Romero March (Universidad de las Islas Baleares, España), al Dr. Juan Pedro Montávez Gómez (Universidad de Murcia, España), al Sr. Jesús Fernández Fernández (Universidad del País Vasco, España) y a nuestro compañero de departamento Sr. Salvador José Ribas Rubio.

Barcelona, mayo de 2004.

Antonio Barrera, Miguel Ángel Prat y María del Carmen Llasat.

Grupo de Análisis de Situaciones Meteorológicas Adversas (GAMA)  
Departamento de Astronomía y Meteorología.  
Facultad de Física. Universidad de Barcelona.  
Av. Diagonal 647, E-08028 BARCELONA (España)

<http://gama.am.ub.es/>

Tel: +34 934021124

Fax: +34 934021133

e-mail: [tbarrera@am.ub.es](mailto:tbarrera@am.ub.es), [maprat@am.ub.es](mailto:maprat@am.ub.es), [carmell@am.ub.es](mailto:carmell@am.ub.es)



## 1.1 Requerimientos de hardware y software

En este apartado se detalla cuál es el hardware y software necesario para instalar el modelo, así como de dónde obtenerlo.

### 1.1.1 LINUX

La versión Red Hat 9 de Linux se puede descargar desde:

- <http://ftp.redhat.com/pub/redhat/linux/9/en/iso/i386/>

Hay que descargar en 3 CD's estos tres ficheros **.iso**:

**shrike-i386-disc1.iso**  
**shrike-i386-disc2.iso**  
**shrike-i386-disc3.iso**

Unas instrucciones para la instalación del Red Hat 9 se pueden descargar desde:

- [http://www4.terra.com.ar/canales/cgi-bin/imprimir.pl?noticia\\_id=72061&canal\\_id=6](http://www4.terra.com.ar/canales/cgi-bin/imprimir.pl?noticia_id=72061&canal_id=6)

### 1.1.2 Compiladores FORTRAN de INTEL (IFC)

Los compiladores FORTRAN se pueden conseguir desde la página de INTEL:

- <http://www.intel.com/software>

o directamente desde:

- [http://downloadfinder.intel.com/scripts-df-external/filter\\_results.aspx?strTypes=all&ProductID=907&OSFullName=Linux\\*&strOSs=39&submit=Go%21](http://downloadfinder.intel.com/scripts-df-external/filter_results.aspx?strTypes=all&ProductID=907&OSFullName=Linux*&strOSs=39&submit=Go%21)

si fuera necesario bajar los de C++, que en principio están incorporados dentro del Red Hat 9, debería hacerse desde:

- [http://downloadfinder.intel.com/scripts-df-external/Product\\_Filter.aspx?ProductID=905](http://downloadfinder.intel.com/scripts-df-external/Product_Filter.aspx?ProductID=905)

La versión del compilador de FORTRAN a escoger es la no comercial, por lo tanto habrá de rellenarse y enviar un formulario, con lo que se recibirá las instrucciones para descargar el compilador en la dirección de correo que se haya indicado en el formulario. Es importante destacar que la versión no comercial es para un **solo usuario**.

Para cualquier duda se puede acceder con el código de licencia a:

- <http://www.intel.com/software/support>



## Manual de instalación del MM5 usando los compiladores FORTRAN de INTEL (IFC)

---

El mensaje recibido llega con las instrucciones para descargarlo e instalarlo; y con un código de licencia.

El programa y la licencia se descarga desde:

- <http://www.intel.com/software/products/compilers/downloads/forlin.htm>

el fichero se llama: **I\_fc\_p\_7.1.008.tar** y contiene el fichero de la licencia

el fichero con la licencia que tiene un nombre como: **I\_for\_45594012.lic**

### 1.1.3 NCAR Graphics

El programa NCARG Graphics es gratuito, solo hay que rellenar una solicitud en la página:

- <http://ngwww.ucar.edu/ng4.2/download.html>

una vez enviada la solicitud, se llega a la página:

- <http://ngwww.ucar.edu/cgi/getsrc.cgi>

pero para llegar a esta página se ha de pasar por la del registro, sino no funciona. Ahí se encontrará con dos versiones: una para instalar desde cero, y la otra con módulos pre-compilados. Es aconsejable descargarse la versión para instalar desde cero. El programa se graba en un CD y no depende al instalarlo del sistema operativo de la máquina.

El fichero a descargar se llama: **4.3.1.source\_code.tar**

Hay instrucciones para instalarlo en la dirección

- <http://ngwww.ucar.edu/ng4.3/installsrc.html>

### 1.1.4 VIS5D+

Este es un programa de libre distribución y para aprovechar todas sus posibilidades se han de descargar los siguientes ficheros:

**vis5d+-1.1.tar.gz**

**vis5d+-1.2.1.tar.gz**

que se pueden descargar desde:

- [http://sourceforge.net/project/showfiles.php?group\\_id=9290](http://sourceforge.net/project/showfiles.php?group_id=9290)

**vis5d-data.tar.Z**

que se descarga desde:



- <http://www.ssec.wisc.edu/~billh/vis5d.html>

**netcdf-3.4.tar.Z**

que se descarga desde:

- <ftp://ftp.unidata.ucar.edu/pub/netcdf/netcdf-3.4.tar.Z>

hay también un fichero de instrucciones llamado:

**README\_vis5D**

## 1.1.5 MM5

El modelo MM5 está compuesto por una serie de ficheros independientes que se han de compilar por separado y que vienen por separado en formato **.TAR.gz**. Los cuales se descargan desde la dirección:

- <http://www.mmm.ucar.edu/mm5/mm5-home.html>

y desde aquí yendo a la opción Download.

MM5V3 – los paquetes

MM5V3 Terrain Data – los datos (se descargan todos)

**TERRAIN.TAR.gz**  
**REGRID.TAR.gz**  
**FETCH.TAR.gz**  
**LITTLE\_R.TAR.gz**  
**INTERPF.TAR.gz**  
**MM5.TAR.gz**  
**INTERPB.TAR.gz**  
**NESTDOWN.TAR.gz**  
**MM5toGrADS.TAR.gz**  
**toVIS5D.TAR.gz**  
/TERRAIN\_DATA  
    **DEM\_02M\_GLOBAL.gz**  
    **LANDUSE-OLD.TAR.gz**  
    **LANDUSE.10**  
    **LANDWATERMASK-SIB.TAR.gz**  
    **LANDWATERMASK-USGS.TAR.gz**  
    **LWMASK-USGS.10 ; LWMASK-USGS.05**  
    **LWMASK-SIB.02.gz**  
    **LWMASK-SIB.30s.gz**  
    **LWMASK-USGS.02.gz**  
    **LWMASK-USGS.30s.gz**  
    **SOILCAT.02.gz**  
    **SOILCAT.30s.gz**  
    **SOILCAT.TAR.gz**  
    **SOILCATB.02.gz**



**SOILCATB.30s.gz**  
**SOILCATB.TAR.gz**  
**SOILTEMP.60**  
**TER.TAR.gz**  
**VEG-FRACTION.10.gz**  
**VEG-SIB.02.gz**  
**VEG-SIB.30s.gz**  
**VEG-USGS.02.gz**  
**VEG-USGS.30s.gz**  
**VEGETATION-SIB.TAR.gz**  
**VEGETATION-USGS.TAR.gz**



## 2. INSTALACIÓN DE LOS REQUERIMIENTOS

En este apartado se explican cómo instalar todo el software necesario para poder instalar correctamente el MM5, así como el sistema operativo Linux, es decir cómo instalar los requerimientos previos que necesita el sistema.

### 2.1 Sistema Operativo LINUX

Se arranca el PC con el CD-1 del Red Hat 9 y con un disco duro sin formatear. A continuación se inicia la instalación de LINUX. Siguiendo las instrucciones siguientes se llega al final sin más problemas, extraídas de la dirección:

- [http://www4.terra.com.ar/canales/cgi-bin/imprimir.pl?noticia\\_id=72061&canal\\_id=6](http://www4.terra.com.ar/canales/cgi-bin/imprimir.pl?noticia_id=72061&canal_id=6)

#### Nunca fue tan fácil instalar Linux

**Salir ileso de una instalación de Linux es el primer gran desafío de todos los que se quieren adentrar en el mundo del software libre y gratuito. En la tercera entrega del manual sobre este sistema operativo, repasamos todo sobre la instalación de Red Hat 8, pantalla por pantalla.**

Decidimos por una distribución de Linux no es nada fácil. Al final nos hemos quedado con Red Hat 8. Red Hat es una de las distribuciones más estables y estándar de las que se pueden encontrar, aunque su instalación no es de las más fáciles, es lo suficiente para que cualquiera lo pueda instalar y configurar.

Una vez registrados en Red Hat Network, su sistema de actualización automático nos mantendrá el sistema a la última. La facilidad de uso de otras distribuciones las hacen más difíciles a la hora de actualizar, o inestables una vez actualizadas, en Red Hat estos problemas son bastante más difíciles de encontrar.

Realizaremos la instalación en modo gráfico. Aunque ello supone unos requerimientos de máquina un poco más altos y más tiempo de instalación, esta será más sencilla y amena de realizar.

#### ¿Con qué computadora puedo instalar Linux?

Como sucede con otros sistemas operativos, Linux necesita de unos requisitos mínimos para poder instalarse y funcionar con mínimo de calidad. Estos requisitos se pueden reducir si tanto la instalación como su uso van a ser únicamente en modo texto. Nosotros vamos a instalarlo y ejecutarlo en modo gráfico, ya que va a ser mucho más sencillo.

	Mínimo	Recomendado
Procesador	Pentium 100 MHz	Pentium II 266 Mhz
Memoria	128 MegaBytes	256 MegaBytes
Disco duro	3 GigaBytes	6 GigaBytes
CD Rom	Necesario (4X)	Necesario (24X)
Tarjeta ráfica	16MegaBytes (VGA)	24MegaBytes (SVGA)





## Manual de instalación del MM5 usando los compiladores FORTRAN de INTEL (IFC)

### Sobre la preinstalación

Para la instalación de Red Hat podemos usar tanto un disco rígido entero para él, como un disco rígido con dos particiones. En cualquiera de los casos, la partición tiene que estar completamente libre, es decir, sin particionar. Para ello podemos usar, tanto las herramientas que vengan con nuestro sistema operativo, como otras como Fdisk, Partition Magic, etc.

También podemos optar por instalar Red Hat en una máquina virtual, como por ejemplo el VMware, con lo cual el rendimiento de nuestro Linux se verá notablemente reducido. Pero por el contrario podremos disfrutar de ambos sistemas operativos simultáneamente y no necesitaremos particionar el disco.

La instalación la realizaremos desde CDs, por lo que es necesario configurar la computadora para que la primera unidad de arranque del sistema sea el CDROM. Este cambio se hace en la BIOS del computadora, por lo que nos referiremos al manual de la placa base para configurarlo correctamente.

### Manos a la obra

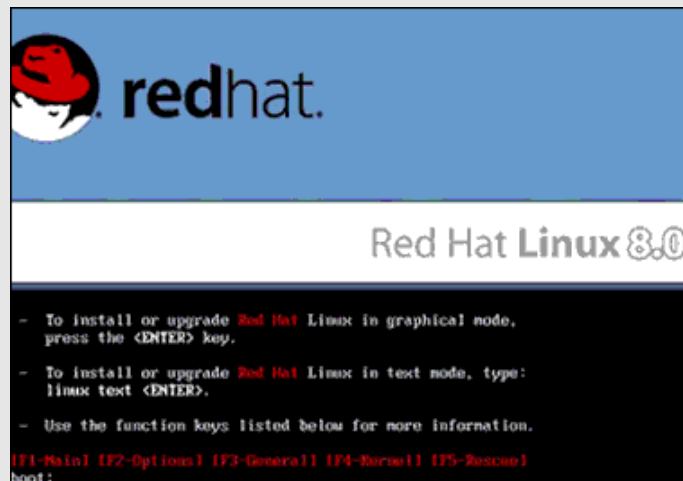
Una vez cumplidos los requisitos de la preinstalación, podemos comenzar la instalación de Red Hat. Para ello introduciremos el primer CD en la unidad lectora y reiniciaremos la computadora. Las primeras pantallas que veremos están en inglés hasta que se selecciona el idioma que se desea para el sistema operativo.

Para hacer más sencillo el seguimiento de las instrucciones, la hemos dividido en los títulos de las pantallas que nos mostrará durante la instalación.

## 1. Inicio

La primera pantalla que nos muestra la instalación es la de Inicio, en la cual podemos seleccionar si queremos realizar la instalación en modo texto o gráfico. Para seleccionar la instalación gráfica simplemente tenemos que apretar la tecla Intro (Enter).

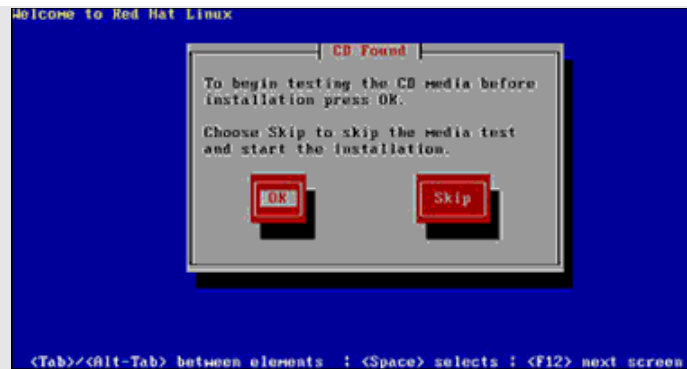
Si esperamos demasiado en apretar la tecla la instalación arrancará automáticamente; pero en modo texto.



## 2. Welcome to Red Hat Linux

Ahora podemos comprobar el estado de los CDs de instalación. Aunque es una tarea lenta, es recomendable hacerlo la primera vez para que no nos deje la instalación a medias.

Para eso simplemente hay que hacer clic en la tecla Espacio. Y si queremos saltar esta comprobación, en F12.



### 3. Welcome

Empezamos la instalación gráfica con la pantalla de bienvenida a Red Hat Linux 8. A partir de este momento ya podemos usar el mouse. Hacemos clic en Next para empezar la instalación.



### 4. Language Selection (selección de lenguaje)

La selección del idioma que se usará para la instalación. En nuestro caso seleccionamos Spanish (Español) y hacemos clic en Next.



## 5. Teclado

La configuración del idioma del teclado es importante, si no lo hiciésemos no funcionarían los caracteres del castellano como la "ñ" o los acentos.

Seleccionamos Spanish y hacemos clic en Siguiente.



## 6. Configuración del mouse

Ahora toca seleccionar y configurar el mouse. Por lo general el sistema selecciona automáticamente el más similar al que tenemos instalado.

Podemos probar de navegar por los diferentes tipos a ver si encontramos uno más adecuado. Una vez seleccionado tecleamos sobre Siguiente.



## 7. Tipo de instalación

En esta pantalla tendremos que seleccionar el tipo de instalación que deseamos realizar. De esta selección dependerá el uso que le daremos al sistema operativo y los "paquetes" de utilidades que se instalarán. Marcaremos Estación de Trabajo, con esta configuración tendremos los paquetes más comunes para trabajar, y haremos clic en Siguiente.

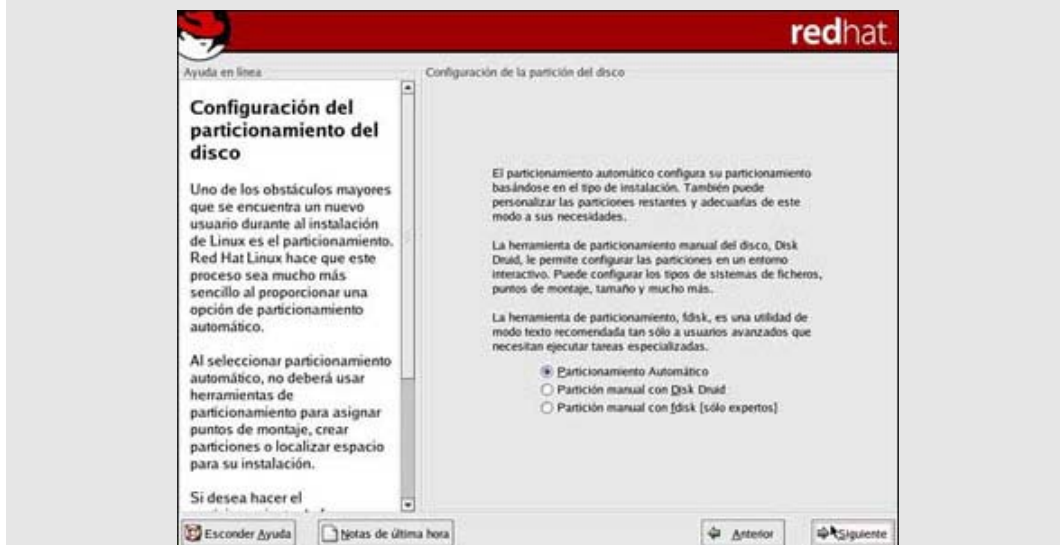
De todas formas más adelante podremos seleccionar o deseleccionar más componentes.

## Manual de instalación del MM5 usando los compiladores FORTRAN de INTEL (IFC)



### 8. Configuración de la partición del disco

En este paso creamos el sistema de archivos para nuestro nuevo sistema operativo, las particiones. La opción recomendada es el Particionamiento Automático. Con esta opción el sistema realizará todas las modificaciones por nosotros. En el caso de disponer de varias unidades de disco, en la siguiente pantalla podremos seleccionar en que unidad queremos crear las particiones. Las opciones de Disk Druid y Fdisk solo son recomendables cuando se tiene experiencia con el sistema de archivos de Linux. Por lo tanto dejamos marcada la opción por defecto y hacemos clic en Siguiente.



### 9. Particionamiento automático

Es una de las fases críticas de la instalación, ya que dependerá del buen uso que hagamos de esta utilidad, el mantener o no otros sistemas operativos que ya se encuentren instalados así como sus datos. Disponemos de tres opciones: Eliminar todas las particiones Linux del disco duro seleccionado, Eliminar todas las particiones, o guardar las particiones existentes y usar solo el espacio libre. La selección de una u otra opción dependerá de si deseamos instalarlo en un único disco rígido, o en un disco rígido con otros sistemas operativos los cuales queremos mantener. En el caso de disponer de **un disco rígido entero para Linux**, seleccionaremos la segunda opción, Eliminar todas las particiones de este sistema, marcaremos el disco duro donde queremos instalarlo, y haremos clic en Siguiente. Si deseamos instalarlo en un disco rígido con **otras particiones**, seleccionaremos la tercera opción, Guarde todas las particiones y use el espacio libre existente, también debemos marcar el disco rígido correspondiente, después hacemos clic en Siguiente.

## Manual de instalación del MM5 usando los compiladores FORTRAN de INTEL (IFC)

### 10. Configuración del gestor de arranque

El gestor de arranque es la utilidad que nos ayudará a elegir entre los sistemas operativos instalados. Por defecto viene el GRUB, mucho más atractivo y flexible que LILO, el otro gestor de arranque de Linux.

También podemos ver si ha detectado otros sistemas operativos, si los tenemos instalados, y seleccionar cual será el sistema operativo que arrancará por defecto. Lo dejamos todo por defecto y hacemos clic en Siguiente.



### 11. Configuración de la red

Aquí es donde especificaremos todas las opciones necesarias para nuestra red, o Router ADSL. Si tenemos una red de varias computadoras, tendremos que especificar los valores necesarios para que funcione; si la dirección IP se asigna manualmente o automáticamente a través de DHCP. Para esta última opción, necesitaremos una computadora en nuestra red que se encargue de asignarnos una dirección IP automáticamente. Algunos Routers de ADSL tienen la opción de asignar direcciones a través de DHCP.

Esta configuración depende de la instalación de red que tenga el usuario, por lo que no hay unos pasos fijos a seguir. Para asignar una IP fija o dinámica simplemente tenemos que hacer clic en el botón Modificar.

En nombre del Host, marcaremos de forma manual, y en el siguiente cuadro de texto escribiremos el nombre que deseamos para este ordenador. Una vez realizados los cambios necesarios hacemos clic en Siguiente.



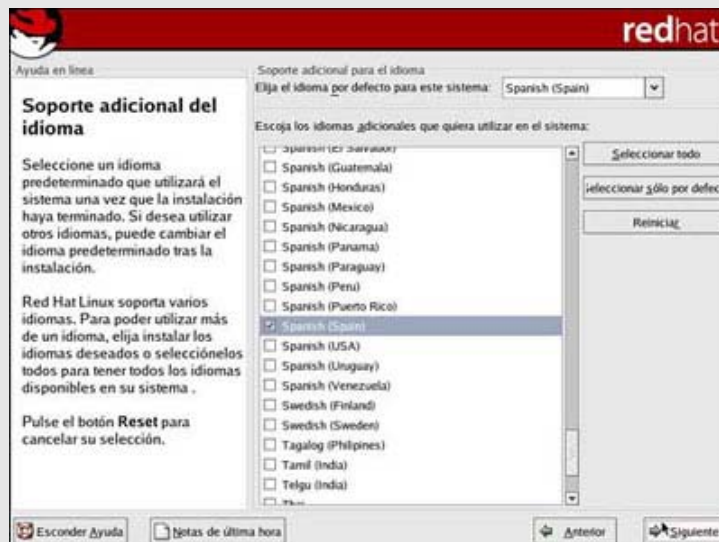
## 12. Configuración del Firewall

El Firewall es el software que evitará las entradas no deseadas desde Internet a nuestro equipo. El nivel de seguridad por defecto es Intermedio. Si nuestra PC no está conectado directamente a Internet, podemos desactivarlo marcando la opción de Ningún Firewall, que sería lo aconsejable si queremos que todo funcione correctamente. Si por el contrario esta conectada a Internet directamente, dejaremos los valores por defecto y marcaremos Usar las reglas del firewall predeterminadas. Una vez efectuadas las selecciones apretamos Siguiente.



## 13. Soporte adicional para el idioma

Este es el idioma con el que arrancará operativo, podemos seleccionar varios y cambiar de idiomas sin necesidad de salir de Linux. Nosotros seleccionaremos Spanish (Spain) y haremos clic en Siguiente.



## 14. Selección del huso horario

Podemos seleccionar nuestra zona horaria entre las muchas que se encuentran en la lista, o seleccionarla manualmente usando UTC.



## 15. Configuración de las cuentas

El usuario "root", es el administrador del sistema Linux. Lo primero que hemos de hacer es asignarle una contraseña. Una vez la contraseña se encuentre confirmada y aceptada, procederemos a la creación de un usuario normal, esto lo haremos haciendo clic en Añadir.

Se nos abre una ventana para definir los parámetros del nuevo usuario, entre los datos que nos solicita están: el nombre del usuario, que es el nombre que introduciremos cada vez que queramos entrar en Linux. La contraseña y su confirmación. Y el nombre completo, que es el nombre que aparecerá como nuestra información.

Una vez rellenados todos los datos hacemos clic en OK para aceptar el nuevo usuario. Si queremos añadir más usuarios simplemente tenemos que repetir la operación cuantas veces deseemos. Una vez introducidos todos los usuarios hacemos clic en Siguiente.



## 16. Estación de trabajo predeterminada

Este paso es donde podemos añadir o eliminar componentes para nuestra instalación. Nosotros seleccionaremos Personalizar el grupo de paquetes a instalar. Y haremos clic en Siguiente.

La siguiente pantalla nos mostrará los paquetes que tenemos seleccionados y los que podemos seleccionar. Como mínimo añadiremos el **Entorno de escritorio para KDE**, esto nos brindará el escritorio KDE que es uno de los más fáciles y flexibles de usar.

No se tiene que eliminar ningún componente, ya que puede resultarnos necesario en algunas ocasiones, o ser una dependencia de otro paquete que si deseamos instalar. En cuanto agregar nuevos grupos de paquetes podemos hacerlo según nuestras necesidades.

Si hacemos clic en el botón Detalles, obtendremos la lista de los paquetes individuales que forman el grupo. Al igual que con los grupos anteriores, no es buena idea quitar ningún elemento, pero podemos

## Manual de instalación del MM5 usando los compiladores FORTRAN de INTEL (IFC)

agregarlos a nuestro gusto.

Una vez acabadas las selecciones hacemos clic en Siguiente.



### 17. Sobre la instalación

Ya está el sistema preparado para la instalación de paquetes, haciendo clic en Siguiente comenzará la copia de archivos al disco rígido.



### 18. Instalando paquetes

El copiado de archivos al disco rígido puede tardar, dependiendo del equipo que dispongamos y las opciones seleccionadas, entre 30 y 90 minutos aproximadamente.

Durante la instalación propiamente dicha, se nos pedirá que insertemos el segundo y tercer CDRom de instalación.





## 19. Creación del disco de arranque

Este disco de arranque nos permitirá arrancar el sistema Linux si por error falla el gestor de arranque GRUB, lo borramos sin querer, o si decidimos prescindir de él.



## 20. Configuración de la interfaz gráfica (X)

Como norma, si disponemos de una tarjeta gráfica más o menos moderna, será detectada por el propio Linux. Si no encuentra nuestra tarjeta, intentará instalar la que más se acerque a sus posibilidades. Por lo general una vez acabada la instalación será necesario instalar los drivers de la tarjeta gráfica, si los hay disponibles para Linux, si queremos sacarle el máximo partido. Dicha instalación se deberá hacer siguiendo las instrucciones del fabricante.

En esta pantalla, si no encontramos nuestra tarjeta gráfica en la lista dejamos la que Linux ha elegido por nosotros y hacemos clic en Siguiente.

## Manual de instalación del MM5 usando los compiladores FORTRAN de INTEL (IFC)



### 21. Configuración del monitor

La configuración del monitor se hace de la misma forma que la configuración de la interfaz gráfica. Si Linux no encuentra un monitor que se ajuste al que tenemos, ni se encuentra uno similar en la lista, dejamos el que Linux nos asigna por defecto. Eso sí, en este último caso, si conocemos tanto la frecuencia horizontal, como la vertical, la ponemos en los correspondientes cuadros de texto, y hacemos clic en Siguiente.



### 22. Personalizar la configuración gráfica

Sólo nos queda decidir la profundidad de color que queremos, la resolución del escritorio y si queremos que arranque Linux en modo gráfico o en modo texto. Lo aconsejable es seleccionar el máximo número de colores y una resolución no inferior a 1024 x 768, y por supuesto el arranque en modo gráfico. Una vez hechas las modificaciones oportunas hacemos clic en Siguiente.



### 23. Enhorabuena

Terminamos la instalación de Linux y solo nos queda hacer clic en el botón Salir, para que la máquina se reinicie. Si seguimos los pasos correctamente, y no tenemos ningún problema de incompatibilidad con nuestro hardware, Linux arrancará perfectamente.

Lo primero que nos mostrará al arrancar la PC es el **gestor de arranque GRUB**, con el cual seleccionaremos el sistema operativo que queremos ejecutar. Una vez seleccionado Linux empezará a cargar.

Una vez finalizada la carga, veremos la entrada a Red Hat, en donde lo primero que tendremos que hacer es introducir nuestro nombre de usuario y teclear **Intro**, lo siguiente que nos solicita es la contraseña de este usuario, pero antes de dársela vamos a seleccionar el escritorio que queremos por defecto, ya que el predeterminado es GNOME y nosotros queremos arrancar con **KDE**. Para eso hacemos clic en **Sesión**, que nos abre una ventana donde seleccionaremos KDE y OK. Una vez realizado lo anterior ya podemos decirle la contraseña y apretar la tecla Intro

Ya estamos dentro del escritorio KDE de Linux. En este punto arranca un asistente, que nos ayudará a personalizar nuestro nuevo sistema operativo. Ahora ya solo nos queda sacarle **el máximo partido** que

Después de realizar las instrucciones anteriores, arranque el PC, acceda como root y compruebe que tiene instalado correctamente el S.O.

Cree el usuario mm5i, indicando para el mismo la shell csh

Cree la carpeta /home/mm5i/bin

Reinicie el sistema.



## 2.2 Esquema general

Para la instalación del software requerido hay que tener en cuenta que se van a utilizar dos usuarios:

- **root:** el administrador del equipo, con todos los privilegios para instalar, copiar, escribir y ejecutar programas o ficheros.
- **mm5i:** usuario que utilizará el equipo para correr el modelo MM5. Como hay paquetes con licencias para un solo usuario, se seguirá el criterio que los paquetes que se instalen, que son para correr el MM5, el usuario sea el mm5i. Por tanto aparecerá así en los permisos, rutas, etc...

Para correr el programa MM5 es necesario que se tengan determinadas las shells del sistema y del usuario a c-shell. Para ello se inicia la sesión como root y se edita el fichero **passwd** que se encuentra en **/etc**:

```
#su root  
#cd /etc  
#emacs passwd &
```

debe ponerse en las líneas correspondientes a las shells utilizadas por el usuario y el sistema lo siguiente (en negrita):

```
root:x:0:0:root:/root:/bin/csh  
usuario:x:ID_usr:ID_grupo:nombre_completo_usr:/home:/bin/csh
```

home es la ruta del usuario que va a utilizar el MM5.

- **Ordenación de los paquetes**

Recomendamos que se instalen todos los paquetes del MM5 dentro de un mismo directorio llamado **/home/mm5i/mm5v3**.

La estructura final de la instalación recomendamos que sea la que podemos ver en el siguiente esquema (**Figura 1**) aunque según los conocimientos del usuario puede variar.

Manual de instalación del MM5 usando los compiladores  
FORTRAN de INTEL (IFC)

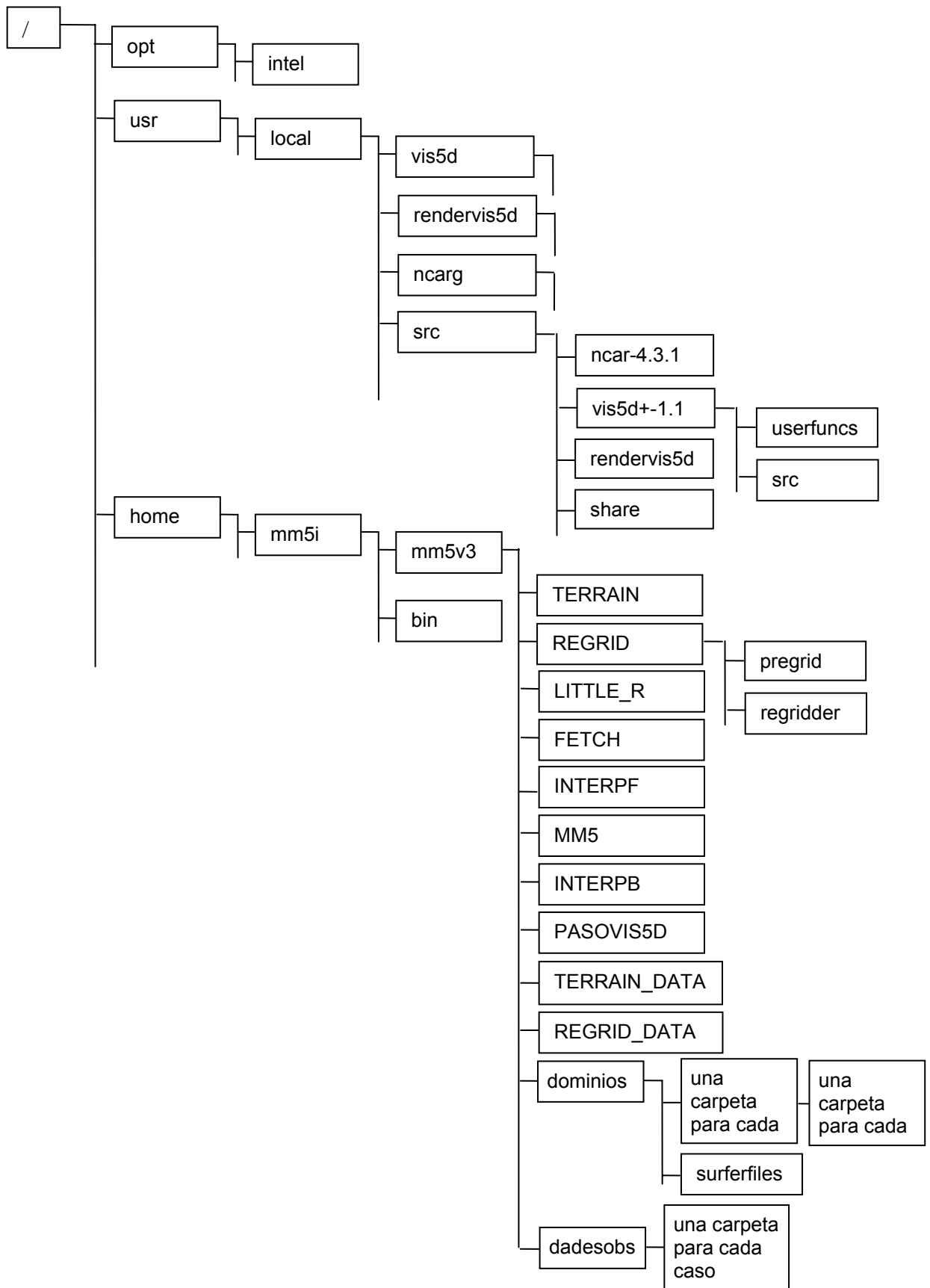


Figura 1. Esquema general de la distribución de todos los elementos necesarios para el MM5



## 2.3 Compiladores FORTRAN

Arranque la sesión de Linux y entre como root. Abra una terminal, copie los ficheros del compilador en una carpeta temporal y descompacte ahí el fichero **I\_fc\_p\_7.1.008.tar**. Coloque el fichero de licencia (**I\_for\_45594012.lic**) en la carpeta de instalación **/opt/intel/licenses**:

```
#cp I_for_45594012.lic /opt/intel/licenses
```

Ejecute la instalación:

```
#./install
```

escoja las siguientes opciones:

```
IA32
fortran
directorio por defecto
opción (-U) por defecto
```

y cuando termina, salga del proceso de instalación con la opción

```
- x
```

Los compiladores se instalan en la carpeta:

```
"/opt/intel"
```

y la licencia en la carpeta:

```
"/opt/intel/licenses"
```

Como la licencia no comercial (licencia gratuita) es para un único usuario, la carpeta **intel** e inferiores deben ser propiedad del usuario que va a utilizar el compilador. Para ello haga:

```
#su root
#cd /opt
#chown -R mm5i:mm5i intel
```

Tendrá también que editar el fichero **.cshrc** del usuario, si no existe, copie el **.bashrc** con dicho nombre, luego proceda a borrar el contenido original y añada las dos líneas siguientes para hacer accesible el compilador, debe quedar como lo siguiente:

```
#! bin/csh
```

```
# PATH principal
```

```
setenv PATH /usr/local/sbin:/opt/intel/compiler70/ia32/bin:/usr/sbin:/sbin:${PATH}:${HOME}/bin
```

```
# Nuevos alias
```



## Manual de instalación del MM5 usando los compiladores FORTRAN de INTEL (IFC)

---

```
alias ls 'ls -l --color'  
alias cd "cd !*;pwd"  
alias pwd 'echo $cwd'
```

# Definiciones del compilador Intel

```
source /opt/intel/compiler70/ia32/bin/ifcvars.csh
```

Ahora cambie los permisos para que acceda el usuario mm5i:

```
#chown mm5i:mm5i .cshrc
```

### 2.4 NCAR Graphics

Recomendamos también que se instale el NCAR Graphics. Éste es un software semi-gratuito muy útil para el paquete TERRAIN, sobretodo a la hora de escoger los diferentes dominios y posición de los anidamientos. La parte del programa que utiliza el módulo TERRAIN es gratuita.

Cree la carpeta **/usr/local/ncarg-4.3.1**:

```
#mkdir /usr/local/ncarg-4.3.1
```

copie y descompacte el fichero **4.3.1.source\_code.tar** en la carpeta **/usr/local/ncarg**, ya que ésta es la ruta donde el módulo TERRAIN va a buscar el NCAR Graphics para su ejecución.

Es importante definir en el fichero **.cshrc** del usuario que va a utilizar el MM5 la ruta donde se encuentra instalado el NCAR Graphics. Edite el fichero **/home/mm5i/.cshrc**, el cual debe quedar:

```
#! bin/csh
```

```
# PATH principal
```

```
setenv NCARG_ROOT /usr/local/ncarg
```

```
setenv PATH $NCARG_ROOT/bin:$PATH
```

```
setenv MANPATH $NCARG_ROOT/man
```

```
setenv PATH /usr/local/sbin:/opt/intel/compiler70/ia32/bin:/usr/sbin:/sbin:${PATH}:${HOME}/bin
```

```
# Nuevos alias
```

```
alias ls 'ls -l --color'
```

```
alias cd "cd !*;pwd"
```

```
alias pwd 'echo $cwd'
```

```
# Definiciones del compilador Intel
```

```
source /opt/intel/compiler70/ia32/bin/ifcvars.csh
```

Ahora ejecute:

```
#!/Configure
```

durante la instalación se escogen las opciones por defecto; excepto para la opción de "Acceso a la red para buscar librerías de raster" que se le indica que no

cuando termine el proceso anterior, haga:

```
#make Everything >& make-output &
```

siendo el fichero **make-output** el fichero donde se tendrá el resultado de la compilación.

Ahora borre la carpeta **/usr/local/ncarg-4.3.1**

**NOTA:** Es importante que el propietario de **/usr/local/ncarg** sea el usuario que va a utilizar el MM5, si no es así, haga lo siguiente:

```
#cd /usr/local  
#chown -R mm5i:mm5i ncarg
```

Cierre la sesión y acceda como mm5i.

Ahora haga una prueba con el fichero de test ejecutando desde cualquier carpeta, para comprobar que se ha instalado correctamente el programa, lo siguiente:

```
#ncargex cpex08  
#ctrans -d X11 cpex08.ncgm
```

Esto tiene que representarnos una ventana en blanco y al pulsar el botón derecho del ratón, sobre esta ventana, debe aparecer una imagen dentro de una ventranca X11 (Figura 2). Si esto aparece, el paquete se ha instalado correctamente.

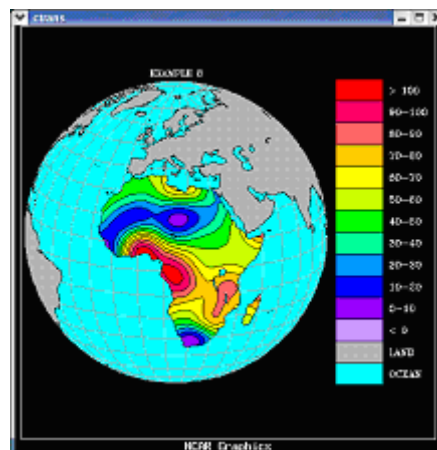


Figura 2. Representación con el NCAR Graphics del fichero para comprobar su correcta instalación.





## 2.5 VIS5D+

El programa Vis5D dispone de varias versiones, habiendo serias diferencias entre ellas. Las posibilidades del programa son muchas pero no todas son posibles implementarlas con la misma versión. Es interesante utilizar las dos versiones debido a las diferentes posibilidades que ofrecen ambas versiones:

- como visor de imágenes MM5 - versiones 1.1 y 1.2.1
- capaz de utilizar funciones externas (programillas para calcular variables de diagnóstico, de inversión de vorticidad o de conversor de datos a otros programas gráficos como es el SURFER) - versión 1.1
- como generador de imágenes vrml, gif, ... de la salida del MM5 - versión 1.2.1

A continuación se detalla cómo instalar cada versión. Según cada caso particular no será preciso hacerlo.

### 2.5.1 VIS5D+ (ver-1.1)

Para instalar esta versión primero hay que convertirse en root y realizar los siguientes pasos:

Definir lo siguiente en el archivo **.cshrc** del usuario mm5i:

```
setenv Vis5D_ROOT /usr/local/vis5d
setenv PATH $Vis5D_ROOT/bin:$PATH
```

Descompactar el fichero vis5d+-1.1.tar.gz en:

```
#cd /usr/local/src
#gunzip vis5d+-1.1.tar.gz
#tar -xvf vis5d+-1.1.tar
```

Entrar en el directorio **visd5d+-1.1** que se ha creado:

```
#cd vis5d+-1.1
```

y descompactar ahí el fichero **vis5d-data-tar.Z**

```
#gunzip vis5d-data-tar.Z
#tar -xvf vis5d-data-tar
```

Descompactar, desagrupar y montar en una carpeta temporal el fichero **netcdf-3.4.tar.z** de la siguiente manera:

```
#gunzip netcdf-3.4.tar.Z
#tar xf netcdf-3.4.tar
#cd netcdf-3.4/src
#./configure
```

## Manual de instalación del MM5 usando los compiladores FORTRAN de INTEL (IFC)

```
#make CC="gcc -Df2cFortran"
```

este proceso acaba con errores, pero no hay que hacerles caso, lo único que se necesita de este archivo, para que funcione la aplicación de userfuncs del Vis5D+, es la librería libnetcdf.a; y esta librería sí que se crea correctamente. Hay que guardarla en el directorio donde se ha descompactado y desagrupado el Vis5D+.

```
#cd libsrc  
#cp libnetcdf.a /usr/local/src/vis5d+-1.1
```

Ejecución de la instalación del programa:

```
#cd /usr/local/src/vis5d+-1.1  
#./configure -prefix=/usr/local/vis5d  
#make
```

Visualización de un fichero de prueba (**hole.v5d**) para comprobar que todo el proceso anterior ha sido correcto:

```
#src/vis5d hole.v5d -path src
```

aparecerá entonces una imagen como la de la **Figura 3**, si se seleccionan los comandos que aparecen iluminados en la imagen.

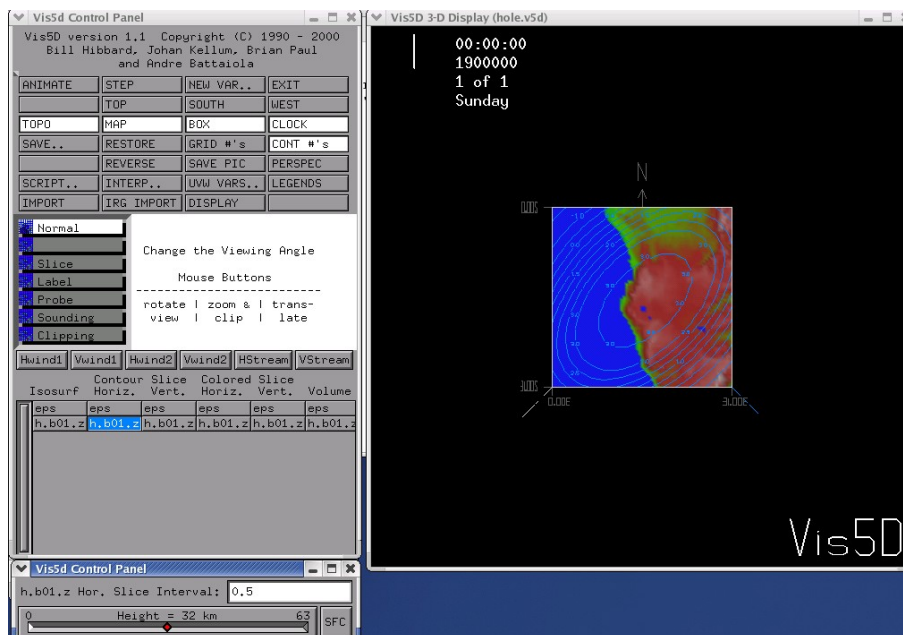


Figura 3. Imagen de ejemplo al visualizar con el Vis5D el fichero hole.v5d

A continuación, teclear lo siguiente, para continuar con la instalación:

```
#make install
```

Visualizar un fichero de prueba (**LAMPS.v5d**) para comprobar que todo el proceso anterior ha sido correcto:

```
#cd src
#vis5d LAMPS.v5d
```

aparecerá entonces una imagen como la de la **Figura 4**, si se seleccionan los comandos que aparecen iluminados en la imagen.

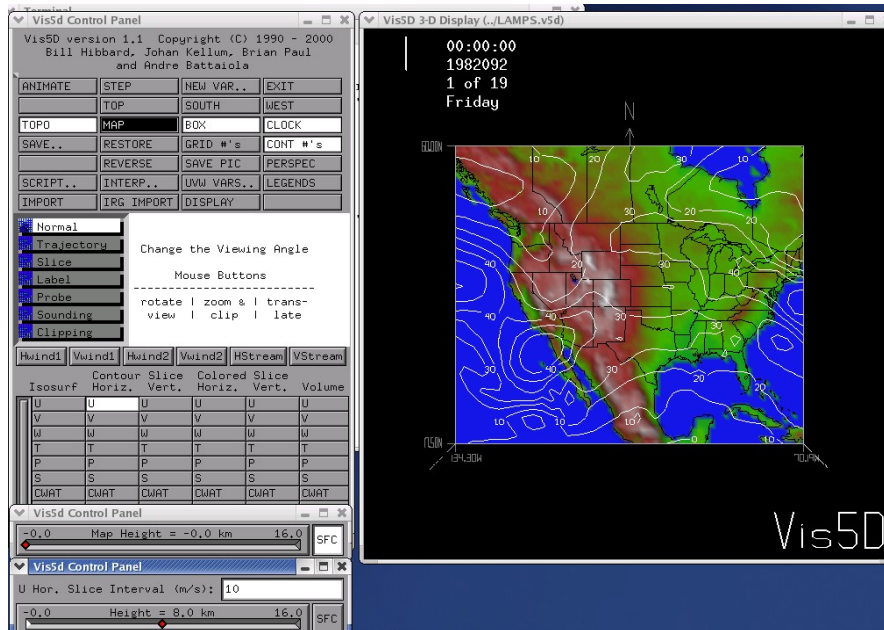


Figura 4. Imagen de ejemplo al visualizar el fichero de prueba LAMPS.v5d

Crear los siguientes links o enlaces que serán necesarios para que funcionen otras aplicaciones para pasar las salidas del MM5 a Vis5D+:

```
#cd /usr/local/vis5d
#ln -s /usr/local/src/vis5d+-1.1/src src
```

### • Uso de funciones externas con Vis5D+ (userfuncs )

El uso de funciones externas es útil para la utilización de funciones creadas por el usuario utilizando los datos del fichero Vis5D, o para otros casos, como se verá más adelante, en la utilización del representador gráfico SURFER.

Para ello se deben modificar los ficheros **externf** y **externf.m**, dentro de la subcarpeta del Vis5d+-1.1 **userfuncs**, para que funcionen correctamente la aplicación de las funciones externas:

```
#cd /usr/local/src/vis5d+-1.1/userfuncs
#emacs externf &
```

cambiar lo que está en negrita:



```
...
    LFLAGS='-nofor_main'
elif [ $id = 'Linux' ] ; then
    # Linux
    CFLAGS='-c -O3'
    FFLAGS='-c -O3'
elif [ $id = 'FOO' ] ; then
    # Other
...

```

en lugar de:

```
...
    LFLAGS='-nofor_main'
elif [ $id = 'Linux' ] ; then
    # Linux
    echo "Fortran needed for external functions"
    exit 1
elif [ $id = 'FOO' ] ; then
    # Other
...

```

**#emacs externf.m &**

cambiar lo que está en negrita:

```
...
# Where to find src/ and util/ directories:
VIS5D_HOME = ../

OBJS = extmain.o socketio.o
#LIBS = $(VIS5D_HOME)util/libmain.a -lm $(EXTLIBS)
LIBS = $(VIS5D_HOME)util/main.o $(VIS5D_HOME)util/doubt.o
$(VIS5D_HOME)util/igmk3d.o $(VIS5D_HOME)util/kludge.o
$(VIS5D_HOME)util/sdest.o $(VIS5D_HOME)util/topo.o -lm $(EXTLIBS)

# Link
...

```

en lugar de:

```
...
# Where to find src/ and util/ directories:
VIS5D_HOME = ../

OBJS = extmain.o socketio.o
LIBS = $(VIS5D_HOME)util/libmain.a -lm $(EXTLIBS)
...

```



## Manual de instalación del MM5 usando los compiladores FORTRAN de INTEL (IFC)

---

Cambiar el propietario de todos los archivos relacionados con el programa Vis5D+ (versión 1.1) de root al usuario del MM5:

```
#cd /usr/local/src
#chown -R mm5i:mm5i vis5d+-1.1
#cd ..
#chown -R mm5i:mm5i vis5d
```

En el caso general para compilar las funciones externas hacer:

```
#cd /usr/local/src/vis5d+-1.1/userfuncs
#./externf nombre_userfunc
```

las userfuncs son ficheros del estilo:

*nombre\_userfunc.f*

**NOTA:** no poner el **.f** en la sentencia de compilación.

Para que aparezcan las userfuncs compiladas al abrir el fichero **.v5d**, hay que crear un link o enlace en el directorio donde está el fichero **.v5d**:

```
#ln -s /usr/local/src/vis5d+-1.1/userfuncs userfuncs
```

**NOTA:** Para visualizar cualquier modelo creado con el vis5d, se recomienda colocar todos los modelos creados en la carpeta **/home/mm5i/mm5v3/dominios/caso de estudio** y luego hacer:

```
#cd /home/mm5i/mm5v3/dominios/caso de estudio
#vis5d nombre_fichero.v5d
```

### 2.5.2 VIS5D+ (ver-1.2.1)

Para instalar esta versión primero hay que convertirse en root y realizar los siguientes pasos:

Definir lo siguiente en el archivo **.cshrc** del usuario mm5i:

```
setenv RenderVis5D_ROOT /usr/local/rendervis5d
setenv PATH $ RenderVis5D_ROOT/bin:$PATH
```

Descompactar el fichero **vis5d+-1.1.tar.gz** en:

```
#cd /usr/local/src
#gunzip vis5d+-1.2.1.tar.gz
#tar -xvf vis5d+-1.2.1.tar
```

Entrar en el directorio **visd5d+-1.2.1** que se ha creado:

```
#cd vis5d+-1.2.1
```

y descompactar ahí el fichero **vis5d-data-tar.Z**

```
#gunzip vis5d-data-tar.Z  
#tar -xvf vis5d-data-tar
```

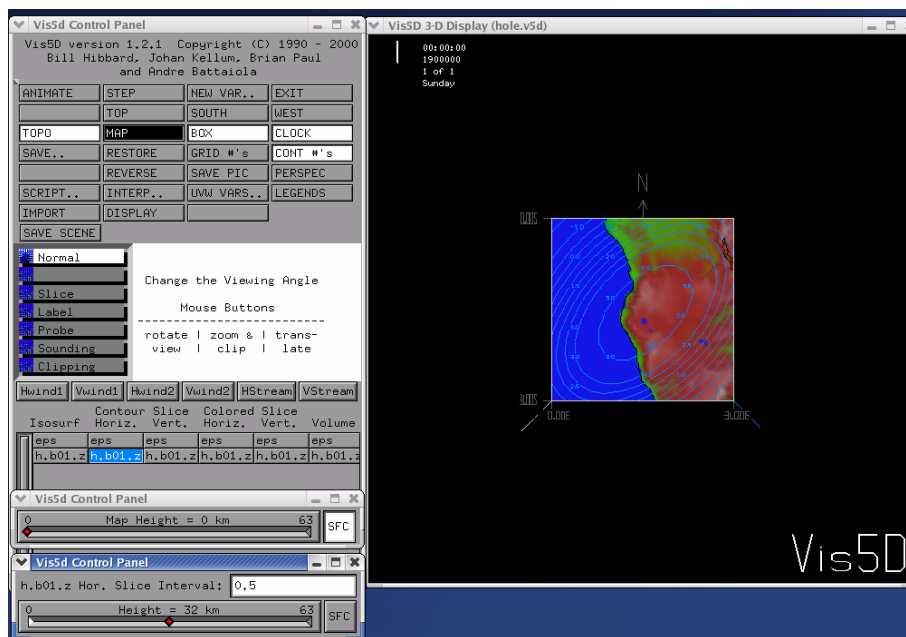
Ejecución de la instalación del programa:

```
#cd /usr/local/src/vis5d+-1.2.1  
#./configure -prefix=/usr/local/rendervis5d  
#make
```

Visualización de un fichero de prueba (**hole.v5d**) para comprobar que todo el proceso anterior ha sido correcto:

```
#src/vis5d hole.v5d -path src
```

y aparecerá entonces una imagen como la de la **Figura 5**, si se seleccionan los comandos que aparecen iluminados en la imagen:



**Figura 5.** Imagen de ejemplo al visualizar con el Vis5D el fichero hole.v5d

A continuación, teclear lo siguiente:

```
#make install
```

Visualizar un fichero de prueba (**LAMPS.v5d**) para comprobar que todo el proceso anterior ha sido correcto:

primero copie el fichero **LAMPS.v5d**

```
#cp /usr/local/src/vis5d+-1.1/LAMPS.v5d /usr/local/src/vis5d+-1.2.1  
#cd src  
#vis5d LAMPS.v5d
```

y aparecerá una imagen tal como la de la Figura 6, si se seleccionan los comandos que aparecen iluminados en la imagen:

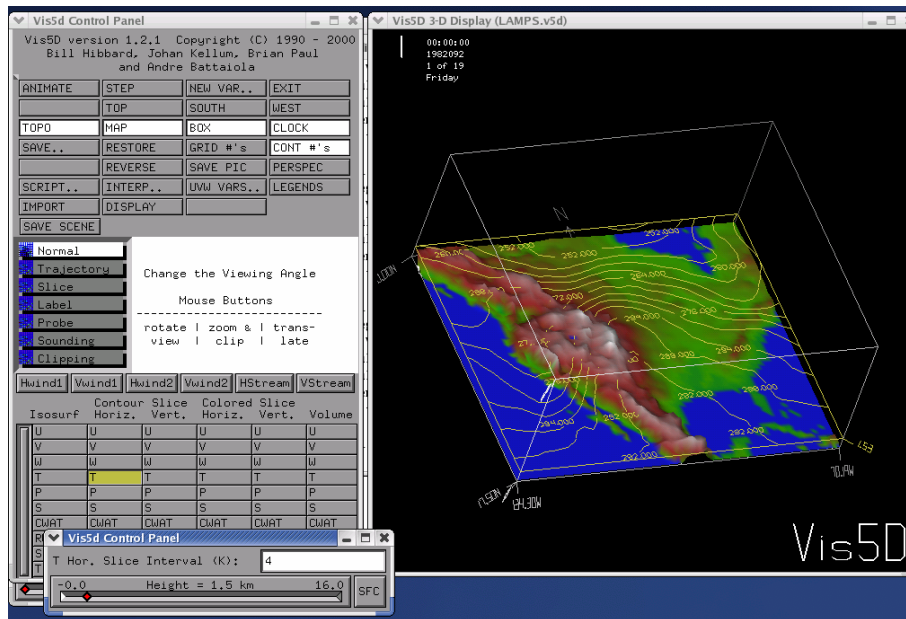


Figura 6. Imagen de ejemplo al visualizar el fichero de prueba LAMPS.v5d

Cambiar el nombre del ejecutable de esta versión del MM5 para no confundirla con la anterior que se ha instalado:

```
#cd /usr/local/rendervis5d/bin  
#mv vis5d rendervis5d
```

Cambiar el propietario de todos los archivos relacionados con el programa Vis5D+ (versión 1.2.1) de root al usuario del MM5:

```
# cd /usr/local/src  
# chown -R mm5i:mm5i vis5d+-1.2.1  
# cd /usr/local/  
# chown -R mm5i:mm5i rendervis5d
```



## 3. INSTALACIÓN DE LOS MÓDULOS DEL MM5

Inicie la sesión como el usuario mm5i. Los ficheros de cada uno de los módulos se copiarán en la carpeta **/home/mm5i/mm5v3** y se descompactarán en carpetas separadas, tal como se indica en los siguientes apartados:

### 3.1 TERRAIN

Es el primer módulo y se descompacta en la carpeta:

**/home/mm5i/mm5v3/TERRAIN**

El fichero con el módulo es:

**TERRAIN.TAR.gz**

Los ficheros de datos son:

**DEM\_02M\_GLOBAL.gz**  
**LANDUSE-OLD.TAR.gz**  
**LANDWATERMASK-SIB.TAR.gz**  
**LANDWATERMASK-USGS.TAR.gz**  
**LWMASK-SIB.02.gz**  
**LWMASK-SIB.30s.gz**  
**LWMASK-USGS.02.gz**  
**LWMASK-USGS.30s.gz**  
**SOILCAT.02.gz**  
**SOILCAT.30s.gz**  
**SOILCAT.TAR.gz**  
**SOILCATB.02.gz**  
**SOILCATB.30s.gz**  
**SOILCATB.TAR.gz**  
**TER.TAR.gz**  
**VEG-FRACTION.10.gz**  
**VEG-SIB.02.gz**  
**VEG-SIB.30s.gz**  
**VEG-USGS.02.gz**  
**VEG-USGS.30s.gz**  
**VEGETATION-SIB.TAR.gz**  
**VEGETATION-USGS.TAR.gz**

Estos ficheros se descompactan y se guardan en la carpeta:

**/home/mm5i/mm5v3/TERRAIN\_DATA**

Ahora, diríjase al directorio **/home/mm5i/mm5v3/TERRAIN** y edite el fichero **Makefile**:

```
#emacs Makefile &
```





Manual de instalación del MM5 usando los compiladores  
FORTRAN de INTEL (IFC)

---

los cambios que deben realizarse, se encuentran en la siguiente parte del archivo (indicados en **negrita**). Este makefile está pensado para utilizar el programa NCAR Graphics:

IFC (Compilador Fortran de Intel):

```
NCARGRAPHICS      =   NCARG  
#NCARGRAPHICS    =   NONCARG
```

```
## Note: Don't forget to remove the libraries (or RHS) in the LOCAL_LIBRARIES  
## line when not using NCAR Graphics.
```

```
INTEL_LIB        =   /opt/intel/compiler70/ia32/lib
```

```
....  
grep Linux .tmpfile ; \  
if [ $$? = 0 ]; then echo "Compiling for Linux" ; \  
  ( $(CD) src ; $(MAKE) all \ \  
    "RM           = $(RM)"      "RM_LIST   = $(RM_LIST)" \ \  
    "LN           = $(LN)"      "MACH      = SGI" \ \  
    "MAKE         = $(MAKE)"    "CPP        = /lib/cpp" \ \  
    "CPPFLAGS     = -I. -P -C -traditional -D$(NCARGRAPHICS)  
    -DRECLENBYTE" \ \  
    "FC          = ifc"         "FCFLAGS = -w90 -w95 -I." \ \  
    "LDOPTIONS   = -i_dynamic"  "CFLAGS  = -I." \ \  
    "LOCAL_LIBRARIES = -L$(NCARG_ROOT)/lib -L/usr/X11R6/lib -  
    lncarg -lncarg_gks -lncarg_c -IX11 -L$(INTEL_LIB) -IPEPCF90  
    -L/usr/lib/gcc-lib/i386-redhat-linux/3.2.2/ -lg2c" ) ; \  
  fi
```

**Nota:** la última librería local no se encuentra siempre en la ruta que se indica, depende de su máquina o sistema operativo. Teclee lo siguiente para ver dónde tiene instalada la librería -lg2c.

```
#gcc -v
```

Este comando devuelve las especificaciones de las librerías gcc y entre ellas se encuentra la de la ruta de la librería lg2c, el mensaje tipo que aparece es del siguiente estilo (en **negrita** aparece la ruta donde están las librerías):

```
Leyendo especificaciones de /usr/lib/gcc-lib/i386-redhat-linux/3.2.2/specs  
Configurado con: ../configure --prefix=/usr --mandir=/usr/share/man --  
infodir=/usr/share/info --enable-shared --enable-threads=posix --disable-checking --  
with-system-zlib --enable-__cxa_atexit --host=i386-redhat-linux  
Modelo de hilos: posix  
gcc versión 3.2.2 20030222 (Red Hat Linux 3.2.2-5)
```



# Manual de instalación del MM5 usando los compiladores FORTRAN de INTEL (IFC)

Lo que aparecía en el fichero Makefile original:

PGF (Compilador Fortran del Portland Group):

```

NCARGGRAPHICS      =      NCARG
#NCARGGRAPHICS     =      NONCARG
## Note: Don't forget to remove the libraries (or RHS) in the LOCAL_LIBRARIES
## line when not using NCAR Graphics.
....
    grep Linux .tmpfile                                     ; \
if [ $$? = 0 ]; then echo "Compiling for Linux"           ; \
( $(CD) src ; $(MAKE) all                                 \
"RM                = $(RM)"      "RM_LIST   = $(RM_LIST) " \
"LN                = $(LN)"      "MACH      = SGI"         \
"MAKE              = $(MAKE)"    "CPP        = /lib/cpp"    \
"CPPFLAGS= -I. -C -traditional -D$(NCARGGRAPHICS) \
-DRECLENBYTE" \
"FC                = pgf90"      "FCFLAGS  = -I. -byteswapio" \
"LDOPTIONS         = "           "CFLAGS   = -I."           \
"LOCAL_LIBRARIES   = -L$(NCARG_ROOT)/lib -L/usr/X11R6/lib \
-lncarg -lncarg_gks -lncarg_c -lX11 -L$(PGI)/linux86/lib -L/usr/lib -lf2c" )
    ; \

```

Dirijase al directorio src

```
#cd src
```

Edite el fichero Makefile

```
#emacs Makefile &
```

Los cambios a realizar se encuentran en la siguiente parte del fichero (en negrita):

```

IFC
all::      terrain.exe data_area.exe rdem.exe rdnml
rdnml:      rdnml.o
             $(FC) rdnml.o -i_dynamic -o $@
terrain.exe: $(OBJS)
             $(FC) -o $@ $(LDOPTIONS) $(OBJS) $(LOCAL_LIBRARIES)

```

En lugar de:

```

PGF
all::      terrain.exe data_area.exe rdem.exe rdnml
rdnml:      rdnml.o
             $(FC) rdnml.o -o $@
terrain.exe: $(OBJS)
             $(FC) -o $@ $(LDOPTIONS) $(OBJS) $(LOCAL_LIBRARIES)

```

Retroceda al directorio TERRAIN:



## Manual de instalación del MM5 usando los compiladores FORTRAN de INTEL (IFC)

---

```
#cd ..
```

Teclee lo siguiente para montar el módulo:

```
#make terrain.deck >& make.out &
```

Edite el fichero **make.out** para comprobar que el proceso de montaje se ha realizado correctamente, y debe aparecer algo similar a:

```
Linux nombre máquina 2.4.20-8 #1 Thu Mar 13 17:54:28 EST 2003 i686 i686 i386  
GNU/Linux  
Making terrain deck for Linux
```

Para ejecutar:

**NOTA:** Para casos reales se ha de editar el fichero **terrain.deck** según las características del caso de estudio, para las particularidades de cada variable y orden dentro de este fichero véase el MM5V3 on-line Tutorial notes:

- <http://www.mmm.ucar.edu/mm5/mm5v3/tutorial/teachyourself.html>

en el caso del módulo de prueba, edite el **terrain.deck** y haga los cambios señalados en negrita:

```
#emacs terrain.deck &
```

### CASO PRUEBA DATOS EN LOCAL

```
#!/bin/csh -f  
# terrain.csh  
#  
set echo  
#  
# Set this if you would like to ftp terrain data  
#  
#set ftpdata = true  
set ftpdata = false  
#  
# Set the following for ftp'ing 30 sec elevation data from USGS ftp site  
#  
#set Where30sTer = ftp  
set Where30sTer = /home/mm5i/mm5v3/TERRAIN_DATA  
if ( $Where30sTer == ftp) then  
#  
....
```



En lugar de:

#### CASO PRUEBA DATOS EN SERVIDOR REMOTO

```
#!/bin/csh -f
# terrain.csh
#
set echo
#
# Set this if you would like to ftp terrain data
#
set ftpdata = true
#set ftpdata = false
#
# Set the following for ftp'ing 30 sec elevation data from USGS ftp site
#
set Where30sTer = ftp
#set Where30sTer = /your-data-directory
if ( $Where30sTer == ftp) then
#
....
```

para ejecutar teclee:

```
#!/terrain.deck >& log &
```

Esto crea un fichero de nombre **log**, donde quedan escritos todos los pasos del proceso de ejecución de TERRAIN, edítelo para comprobar que no ha habido ningún error. Si todo fue correcto debería acabar el fichero con un mensaje tal como:

```
lun dic 29 14:18:43 CET 2003
#!/terrain.exe
rm fort.15 fort.16 fort.18
```

También se crean otros ficheros parecidos al fichero **log** donde se registran los resultados de los diferentes pasos de compilación y ejecución de TERRAIN. El nombre de estos ficheros son:

**terrain.print.out**. Edítelo para comprobar que todo fue bien. Si es así, al final del fichero tendría que aparecer un mensaje tal como:

```
== NORMAL TERMINATION OF TERRAIN PROGRAM ==
FORTRAN STOP 99999
```

**make.terrain.out**. Edítelo para comprobar que todo fue bien. Si es así, al final del fichero tendría que aparecer un mensaje tal como:

```
90 Lines Compiled
/bin/rm -f rdml.f
```



## Manual de instalación del MM5 usando los compiladores FORTRAN de INTEL (IFC)

---

```
ifc rdnm1.o -i_dynamic -o rdnm1
make[1]: Saliendo directorio `/home/mm5i/mm5v3/TERRAIN/src'
```

**make2.print.out.** Edítelo para comprobar que todo fue bien. Si es así, al final del fichero tendría que aparecer un mensaje tal como:

```
59 Lines Compiled
/bin/rm -f write_fieldrec.f
ifc -o terrain.exe -i_dynamic ia.o anal2.o bint.o bndry.o crlnd.o crter.o dfclrs.o exaint.o
finprt.o fudger.o interp.o label.o lakes.o latlon.o llxy.o mxmnl.o nestll.o oned.o outpt.o
output.o pltttr.o rldtr.o replace.o rflp.o setup.o sint.o smth121.o smther.o smthtr.o
terdrv.o terrain.o tfudge.o vtran.o xyobsll.o hiresmap.o plots.o crvst.o crvst30s.o
nestbdy.o crsoil.o equate.o labels.o labelv.o patch.o plotcon.o watercheck.o
crlwmsk.o soil_tg.o water_vfr.o check_data.o terrestrial_info.o write_fieldrec.o -
L/usr/local/ncarg/lib -L/usr/X11R6/lib -lncarg -lncarg_gks -lncarg_c -lX11 -
L/opt/intel/compiler70/ia32/lib -lPEPCF90 -L/usr/lib/gcc-lib/i386-redhat-linux/3.2.2/ -lg2c
/bin/rm -f data_area.exe
ifc -o data_area.exe latlon.o llxy.o mxmnl.o nestll.o rflp.o setup.o outpt.o vtran.o
search.o data30s.o data_area.o -i_dynamic -L/usr/local/ncarg/lib -L/usr/X11R6/lib -
lncarg -lncarg_gks -lncarg_c -lX11 -L/opt/intel/compiler70/ia32/lib -lPEPCF90 -
L/usr/lib/gcc-lib/i386-redhat-linux/3.2.2/ -lg2c
make[1]: Saliendo directorio `/home/mm5i/mm5v3/TERRAIN/src'
```

Si todo el proceso ha sido correcto también se generarán los ficheros, en el caso de dos dominios:

```
TERRAIN_DOMAIN1
TERRAIN_DOMAIN2
TER.PLT
```

Este proceso puede llevar varios minutos, no hacer nada hasta que se termine el proceso. En un P-IV a 2,4 GHz con 512 Mb RAM, tarda aproximadamente unos 7 minutos.

Para comprobar que se han generado correctamente ejecute la instrucción:

```
#!dt TER.PLT
```

y debe aparecer una imagen como la de la **Figura 7**.

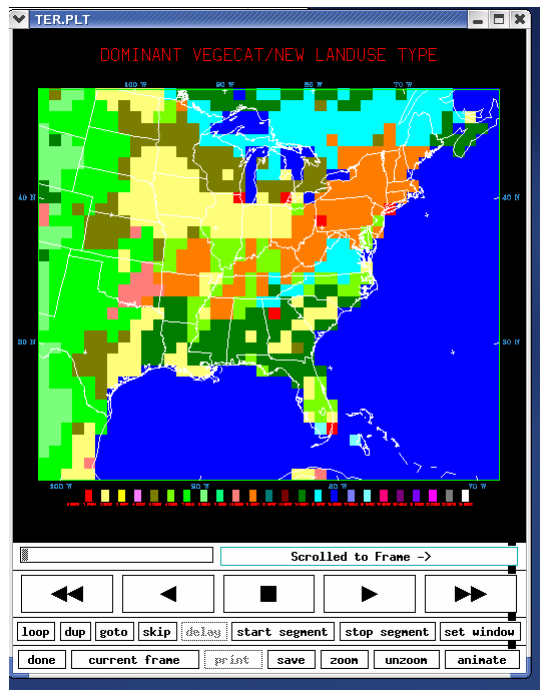


Figura 7. Imagen de visualización del fichero TER.PLT con el NCAR Graphics para el caso de ejemplo (SOC) del manual on-line del MM5.

## 3.2 REGRID

Descompacte el fichero en la carpeta

**REGRID.TAR.gz**

en la carpeta:

**/home/mm5i/mm5v3/REGRID**

Cree la carpeta **/home/mm5i/mm5v3/REGRID\_DATA** y coloque en ella el fichero de prueba:

**NCEP\_ON84.9303**

y en la carpeta **REGRID** edite el fichero **Makefile**:

**#emacs Makefile &**

Los cambios a realizar se encuentran en las siguientes partes del fichero (en negrita):



# Manual de instalación del MM5 usando los compiladores FORTRAN de INTEL (IFC)

IFC:

```
RM_LIST    =    *.o *.M core *.kmo *.mod
INTEL_LIB  =    /opt/intel/compiler70/ia32/lib
```

```
#    Targets for supported architectures
```

....

```
else grep Linux .tmpfile ; \
if [ $$? = 0 ] ; then echo "Compiling for Linux" ; \
    echo "AR" = $(AR)" > macros_pregrid ; \
    echo "RM" = $(RM)" >> macros_pregrid ; \
    echo "RM_LIST" = $(RM_LIST)" >> macros_pregrid ; \
    echo "CD" = $(CD)" >> macros_pregrid ; \
    echo "LN" = $(LN)" >> macros_pregrid ; \
    echo "MAKE" = $(MAKE)" >> macros_pregrid ; \
    echo "SHELL" = /bin/sh" >> macros_pregrid ; \
    echo "TOUCH" = touch" >> macros_pregrid ; \
    echo "CPP" = /lib/cpp" >> macros_pregrid ; \
    echo "CPPFLAGS" = -I. -C -P -DDEC -DBIT32 -traditional"
    >> macros_pregrid ; \
    echo "FC" = ifc" >> macros_pregrid ; \
    echo "FCFLAGS" = -FR -I../util" >> macros_pregrid ; \
    echo "LDFLAGS" = -i_dynamic" >> macros_pregrid ; \
    echo "CCFLAGS" = -DDEC -DBIT32 -I." >> macros_pregrid ; \
    echo "LOCAL_LIBRARIES = ../util/libpgu.a -L$(INTEL_LIB)
    -IPEPCF90" >> macros_pregrid ; \
    ( $(CD) pregrid ; $(MAKE) all ) ; \
    echo "AR" = $(AR)" > macros_regridder ; \
    echo "RM" = $(RM)" >> macros_regridder ; \
    echo "RM_LIST" = $(RM_LIST)" >> macros_regridder ; \
    echo "CD" = $(CD)" >> macros_regridder ; \
    echo "LN" = $(LN)" >> macros_regridder ; \
    echo "MAKE" = $(MAKE)" >> macros_regridder ; \
    echo "SHELL" = /bin/sh" >> macros_regridder ; \
    echo "TOUCH" = touch" >> macros_regridder ; \
    echo "CPP" = /lib/cpp" >> macros_regridder ; \
    echo "CPPFLAGS" = -I. -C -P -DDEC -DBIT32 -traditional"
    >> macros_regridder ; \
    echo "FC" = ifc" >> macros_regridder ; \
    echo "FCFLAGS" = -FR -pc32" >> macros_regridder ; \
    echo "LDFLAGS" = -i_dynamic" >> macros_regridder ; \
    echo "CCFLAGS" = -DDEC -DBIT32 -I." >> macros_regridder ; \
    echo "LOCAL_LIBRARIES = -L$(INTEL_LIB) -IPEPCF90" >>
    macros_regridder ; \
    ( $(CD) regridder ; $(MAKE) all ) ; \
```



## Manual de instalación del MM5 usando los compiladores FORTRAN de INTEL (IFC)

En lugar de:

PGF:

```
RM_LIST = *.o *.M core *.kmo *.mod
```

```
# Targets for supported architectures
```

```
....
```

```
else grep Linux .tmpfile ; \
if [ $$? = 0 ] ; then echo "Compiling for Linux" ; \
echo "AR = $(AR)" > macros_pregrid ; \
echo "RM = $(RM)" >> macros_pregrid ; \
echo "RM_LIST = $(RM_LIST)" >> macros_pregrid ; \
echo "CD = $(CD)" >> macros_pregrid ; \
echo "LN = $(LN)" >> macros_pregrid ; \
echo "MAKE = $(MAKE)" >> macros_pregrid ; \
echo "SHELL = /bin/sh" >> macros_pregrid ; \
echo "TOUCH = touch" >> macros_pregrid ; \
echo "CPP = /lib/cpp" >> macros_pregrid ; \
echo "CPPFLAGS = -I. -C -P -DDEC -DBIT32 -traditional" >>
macros_pregrid ; \
echo "FC = pgf90" >> macros_pregrid ; \
echo "FCFLAGS = -Mfreeform -byteswapio -I./util" >>
macros_pregrid ; \
echo "LDFLAGS = " >> macros_pregrid ; \
echo "CCFLAGS = -DDEC -DBIT32 -I." >> macros_pregrid ; \
echo "LOCAL_LIBRARIES = ../util/libpgu.a" >>
macros_pregrid ; \
( $(CD) pregrid ; $(MAKE) all ) ; \
echo "AR = $(AR)" > macros_regridder ; \
echo "RM = $(RM)" >> macros_regridder ; \
echo "RM_LIST = $(RM_LIST)" >> macros_regridder ; \
echo "CD = $(CD)" >> macros_regridder ; \
echo "LN = $(LN)" >> macros_regridder ; \
echo "MAKE = $(MAKE)" >> macros_regridder ; \
echo "SHELL = /bin/sh" >> macros_regridder ; \
echo "TOUCH = touch" >> macros_regridder ; \
echo "CPP = /lib/cpp" >> macros_regridder ; \
echo "CPPFLAGS = -I. -C -P -DDEC -DBIT32 -traditional" >>
macros_regridder ; \
echo "FC = pgf90" >> macros_regridder ; \
echo "FCFLAGS = -Mfreeform -pc 32 -byteswapio" >>
macros_regridder ; \
echo "LDFLAGS = " >> macros_regridder ; \
echo "CCFLAGS = -DDEC -DBIT32 -I." >> macros_regridder ; \
echo "LOCAL_LIBRARIES = " >> macros_regridder ; \
( $(CD) regridder ; $(MAKE) all ) ; \
```





## Manual de instalación del MM5 usando los compiladores FORTRAN de INTEL (IFC)

También hay que realizar cambios en otros ficheros **Makefile** que se encuentran en las subcarpetas **REGRID/pregrid** y **REGRID/regridder**:

### • pregrid

Vaya a la carpeta **REGRID/pregrid/util** y edite el fichero **Makefile**

```
#cd pregrid
#cd util
#emacs Makefile &
```

Los cambios a realizar son (en negrita):

IFC:

```
gribprint:    gribprint.o libpgu.a
              $(FC) -i_dynamic -o $(@) $(?:.f=.o) $(LOCAL_LIBRARIES)
              $(RM) gribprint.o
```

En lugar de:

PGF:

```
gribprint:    gribprint.o libpgu.a
              $(FC) -o $(@) $(?:.f=.o) $(LOCAL_LIBRARIES)
              $(RM) gribprint.o
```

### • regridder

No hay que realizar ningún cambio aquí.

Para montar el módulo REGRID, vaya a la carpeta **REGRID** y ejecute:

```
#make >& make.out &
```

Edite el fichero **make.out** para comprobar que el proceso de montaje ha sido el correcto. Al final del fichero debería aparecer un mensaje como:

```
420 Lines Compiled
/bin/rm -f regridder.f
ifc -o regridder -i_dynamic proc_grid_store.o proc_ingest_first_guess.o
proc_list_to_array.o proc_make_dot_point_data.o proc_make_big_header.o
proc_make_small_header.o proc_map_to_met_winds.o proc_met_to_map_winds.o
proc_namelist.o proc_output.o proc_tc_bogus.o proc_read_terrain.o
proc_zap_space_array.o proc_zap_space_list.o regridder.o module_constants.o
module_date_pack.o module_diags.o module_file_data.o module_first_guess_data.o
module_gauss.o module_gridded_data.o module_header_data.o
module_horiz_interp.o module_link_list_info.o module_map_utils.o
module_namelist_info.o module_terrain_data.o module_tc_bogus.o module_util.o
make[2]: Saliendo directorio `home/mm5v3/REGRID/regridder/src'
( /bin/rm -f regridder ; ln -s src/regridder . )
make[1]: Saliendo directorio `home/mm5v3/REGRID/regridder'
```



donde home es la ruta del usuario que está utilizando el MM5.

Para ejecutar este paquete hay que realizarlo en dos pasos, primero ejecutar pregrid y luego regridder.

### 3.2.1 Ejecución de pregrid

Dirijase al subdirectorio pregrid y edite el fichero **pregrid.csh** teniendo en cuenta las recomendaciones y explicaciones de la notas del tutorial on-line. En el caso de prueba deberemos realizar las siguientes modificaciones:

```
#cd pregrid
#emacs pregrid.csh &
```

```
set DataDir = /home/mm5i/mm5v3/REGRID_DATA
```

Teclee para ejecutarlo:

```
#!/pregrid.csh >& log &
```

Edite el fichero **log**, para comprobar que la ejecución ha sido la correcta. Si todo ha ido bien debería aparecer al final del fichero un mensaje tal como:

```
*****
Normal termination of program PREGRID_ON84
*****

End of file on C unit 3
mv SNOW:1993-03-13_00 ../ON84_SNOW:1993-03-13_00
mv SNOW:1993-03-13_12 ../ON84_SNOW:1993-03-13_12
mv SNOW:1993-03-14_00 ../ON84_SNOW:1993-03-14_00
```

```
cd home/mm5v3/REGRID/pregrid/on84/..
```

Done with ON84 processing for type SNOW

```
#####
#####
#####
#####
#####
```

Processing for SourceType = SOIL

```
#####
#####
#####
#####
#####
```

donde home es la ruta del usuario que está utilizando el MM5.



y aparecerán 7 ficheros nuevos del tipo **ON84xxxx**

### 3.2.2 Ejecución de regridder

Dirijase al subdirectorio **regridder** y edite el fichero **namelist.input** teniendo en cuenta las recomendaciones y explicaciones de la notas del tutorial on-line. Para el caso de prueba no se realiza ninguna modificación.

```
#cd ..  
#cd regridder  
#emacs namelist.input &
```

Teclee para ejecutarlo:

```
#./regridder >& log &
```

Edite el fichero **log**, para comprobar que la ejecución ha sido correcta. Si todo ha ido bien debería aparecer al final del fichero un mensaje como:

```
*****  
                Attention RAWINS users!  
Here is a handy PARAMETER statement you can use for RAWINS.  
Increase the LMX value by the number of new pressure levels that you want RAWINS  
to add.  
    PARAMETER ( IMX = 35, JMX = 41, LMX = 21)  
*****
```

```
Ending current time:      1993-03-14_00:00:00
```

**NOTA:** Si se va a utilizar el módulo RAWINS es importante tener en cuenta los valores de IMX, JMX y LMX que pone, para ponerlos en el fichero **rawins.deck**, si no se ponen estos valores el módulo no se ejecuta correctamente.

Al final del proceso se genera el fichero **REGRID\_DOMAIN1**

## 3.3 RAWINS

Este paquete se ha conseguido montarlo y ejecutarlo con éxito para el caso de prueba (SOC), pero para casos particulares en los que se utilizan los datos observacionales ADP convencionales no se ha conseguido ejecutarlo con éxito, apareciendo un error de memoria, quizás debido al propio compilador FORTRAN de INTEL. Ante esto se ha optado por utilizar la versión alternativa al RAWINS, el módulo LITTLE\_R, que requiere de la ejecución previa del módulo FETCH, que convierte el formato de los datos observacionales ADP de ON29 a ASCII, formato de entrada de LITTLE\_R. De todas formas, a continuación se explica el procedimiento que sí que ha funcionado para los datos ADP del caso de prueba (SOC).



Descompacte el fichero:

**RAWINS.TAR.gz**

en la carpeta:

**/home/mm5i/mm5v3/RAWINS**

Diríjase a la carpeta anterior y edite el fichero **Makefile** con los cambios a realizar que se encuentran en las siguientes partes del fichero (en negrilla):

IFC:

**NCARGGRAPHICS = NCARG**  
**#NCARGGRAPHICS = NONCARG**  
**#NETCDFOPT = NETCDF**  
**NETCDFOPT = NONETCDF**  
**INTEL\_LIB = /opt/intel/compiler70/ia32/lib**

...

```
grep Linux .tmpfile ;\  
if [ $$? = 0 ]; then echo "Compiling for Linux" ;\  
  ( $(CD) src ; $(MAKE) all \  
    "RM = $(RM)" "RM_LIST = $(RM_LIST)" \  
    "LN = $(LN)" "MAKE = $(MAKE)" \  
    "OPTIONS = -DDEC -DBIT32 -D$(NCARGGRAPHICS) \  
    -D$(NETCDFOPT)" \  
    "CPP = /lib/cpp" "CPPFLAGS = -I. -C -P \  
    -traditional " \  
    "FC = ifc" "FCFLAGS = -I. -pc32" \  
    "LDOPTIONS = -pc32" "CFLAGS = -I." \  
    "LOCAL_LIBRARIES = -L$(NCARG_ROOT)/lib -L/usr/X11R6/lib \  
    -lncarg -lncarg_gks -lncarg_c -lX11 -L$(INTEL_LIB) -lPEPCF90 \  
    -L/usr/lib/gcc-lib/i386-redhat-linux/3.2.2/ -lg2c" ) ;\  
  ;\  
fi
```



# Manual de instalación del MM5 usando los compiladores FORTRAN de INTEL (IFC)

En lugar de:

PGF:

```

NCARGGRAPHICS = NCARG
#NCARGGRAPHICS = NONCARG
#NETCDFOPT = NETCDF
NETCDFOPT = NONETCDF
...
grep Linux .tmpfile ; \
if [ $$? = 0 ]; then echo "Compiling for Linux" ; \
( $(CD) src ; $(MAKE) all \
"RM = $(RM)" "RM_LIST = $(RM_LIST)" \
"LN = $(LN)" "MAKE = $(MAKE)" \
"OPTIONS = -DDEC -DBIT32 -D$(NCARGGRAPHICS) \
-D$(NETCDFOPT)" \
"CPP = /lib/cpp" "CPPFLAGS = -I. -C -P \
-traditional" \
"FC = pgf90" "FCFLAGS = -I. -byteswapio -pc 32" \
"LDOPTIONS = -pc 32" "CFLAGS = -I." \
"LOCAL_LIBRARIES = -L$(NCARG_ROOT)/lib -L/usr/X11R6/lib - \
lncarg -lncarg_gks -lncarg_c -lX11 -L$(PGI)/linux86/lib -L/usr/lib \
-lf2c" ) ; \

```

Diríjase al directorio **src** y edite el fichero **Makefile**, añadiendo lo que está en **negrita**:

IFC:

```

all: rawins.exe

rawins.exe: $(OBJS)
$(FC) -i_dynamic -o $@ $(OBJS) $(LDOPTIONS)
$(LOCAL_LIBRARIES) $(LDLIBS)

code: $(SRC)

```

En lugar de:

PGF:

```

all: rawins.exe

rawins.exe: $(OBJS)
$(FC) -o $@ $(OBJS) $(LDOPTIONS) $(LOCAL_LIBRARIES)
$(LDLIBS)

code: $(SRC)

```

Suba al directorio RAWINS y teclee lo siguiente para montar el módulo:

```
#make rawins.deck >& make.out &
```



## Manual de instalación del MM5 usando los compiladores FORTRAN de INTEL (IFC)

---

Edite el fichero **make.out** para comprobar si el proceso de montaje ha sido el correcto. Si es así, debería aparecer al final del fichero un mensaje tal como:

```
Linux nombre maquina 2.4.20-8 #1 Thu Mar 13 17:54:28 EST 2003 i686 i686 i386
GNU/Linux
Making rawins deck for Linux
```

Edite el fichero rawins.deck siguiendo las instrucciones del tutorial on-line del MM5.

Para ejecutar el módulo, teclee:

```
#!/rawins.deck >& log &
```

Esto crea un fichero de nombre **log**, donde quedan escritos todos los pasos del proceso de compilación para la ejecución de RAWINS, edítelo para comprobar que no ha habido ningún error. Si todo ha sido correcto debería acabar el fichero con un mensaje del estilo:

```
Linux ALFA 2.4.20-8 #1 Thu Mar 13 17:54:28 EST 2003 i686 i686 i386 GNU/Linux
Compiling for Linux
make[1]: Cambiando a directorio `home/mm5v3/RAWINS/src'
make[1]: No se hace nada para `all'.
make[1]: Saliendo directorio `home/mm5v3/RAWINS/src'
attempting to acquire 1 RAOB sounding file(s)
In: `raobsA': El fichero existe
attempting to acquire 1 6-hourly surface analysis input file(s)
In: `sfc6hrA': El fichero existe
attempting to acquire 1 3-hourly surface analysis input file(s)
In: `sfc3hrA': El fichero existe
vie dic 5 12:07:30 CET 2003
vie dic 5 12:07:41 CET 2003
```

donde home es la ruta del usuario que está utilizando el MM5.

También se crea el fichero rawins.print.out, donde también figuran los pasos realizados por RAWINS. Si el proceso de ejecución es correcto debería aparecer al final del fichero un mensaje como:

```
SUCCESSFUL COMPLETION OF PROGRAM RAWINS
STOP 99999
FORTRAN STOP 99999
```

### 3.4 FETCH

Este módulo contiene el script **adp\_to\_little-r.csh** que convierte los datos observacionales ADP en formato ON29 a formato ASCII, formato de entrada necesario para ejecutar el módulo LITTLE\_R.

Descompacte el fichero:



## FETCH.TAR.gz

en la carpeta:

**/home/mm5i/mm5v3/FETCH**

De la siguiente dirección:

- <http://www.mmm.ucar.edu/mm5/mmv5v3/data/wherev3-data.html>

bájese el fichero:

## input2rawins.tar.gz

descompacte este fichero en la carpeta **/home/mm5i/dadesobs/prueba**, obteniendo los siguientes ficheros:

**RAOBS\_A**  
**SFC6HG\_A**  
**SFC3HR\_A**

para realizar la ejecución de este módulo para el caso de prueba del Tutorial on-line (SOC).

Una vez tenga los ficheros situados en sus carpetas, diríjase a la subcarpeta **adp\_sfc** y edite el fichero **Makefile** con los siguiente cambios (en negrita):

```
#cd adp_sfc  
#emacs Makefile &
```

IFC:

```
grep Linux .tmpfile ; \  
if [ $$? = 0 ]; then echo "Compiling for Linux" ; \  
    $(MAKE) target \  
    "MACH           = DEC" \  
    "WLENGTH        = BIT32" \  
    "FC              = ifc" \  
    "FCFLAGS        = -FR" \  
    "CC              = cc" \  
    "CFLAGS         = " \  
    "LDOPTIONS      = -i_dynamic -Vaxlib" \  
    "LIBS           = " ; \  
....
```

```
target:          adp_sfc.exe  
adp_sfc.exe:    $(OBJS)  
                $(FC) $(LDOPTIONS) -o $@ $(OBJS) $(LIBS)  
clean:
```



## Manual de instalación del MM5 usando los compiladores FORTRAN de INTEL (IFC)

---

En lugar de

PGF:

```
grep Linux .tmpfile ; \
if [ $$? = 0 ]; then echo "Compiling for Linux" ; \
    $(MAKE) target \
    "MACH          = DEC" \
    "WLENGTH       = BIT32" \
    "FC             = pgf90" \
    "FCFLAGS        = -Mfreeform -byteswapio" \
    "CC             = cc" \
    "CFLAGS         = " \
    "LDOPTIONS      = " \
    "LIBS           = "; \
....

target:          adp_sfc.exe
adp_sfc.exe:     $(OBJS)
                 $(FC) -o $@ $(LDOPTIONS) $(OBJS) $(LIBS)

clean:
```

Diríjase a la subcarpeta **adp\_upa** y edite el fichero **Makefile** con los siguiente cambios (en negrita):

```
#cd ..
#cd adp_upa
#emacs Makefile &
```

IFC:

```
grep Linux .tmpfile ; \
if [ $$? = 0 ]; then echo "Compiling for Linux" ; \
    $(MAKE) target \
    "MACH          = DEC" \
    "WLENGTH       = BIT32" \
    "FC             = ifc" \
    "FCFLAGS        = -FR" \
    "CC             = cc" \
    "CFLAGS         = " \
    "LDOPTIONS      = -i_dynamic -Vaxlib" \
    "LIBS           = "; \
....

tarjet:          adp_upa.exe
adp_upa.exe:     $(OBJS)
                 $(FC) $(LDOPTIONS) -o $@ $(OBJS) $(LIBS)

clean:
```





En lugar de

PGF:

```
grep Linux .tmpfile ; \  
if [ $$? = 0 ]; then echo "Compiling for Linux" ; \  
    $(MAKE) target \  
    "MACH          = DEC" \  
    "WLENGTH       = BIT32" \  
    "FC            = pgf90" \  
    "FCFLAGS       = -Mfreeform -byteswapio" \  
    "CC            = cc" \  
    "CFLAGS        = " \  
    "LDOPTIONS     = " \  
    "LIBS          = " ; \  
....  
  
target:      adp_upa.exe  
adp_upa.exe: $(OBJS) \  
             $(FC) -o $@ $(LDOPTIONS) $(OBJS) $(LIBS)  
  
clean:
```

Suba al directorio **FETCH** y edite el fichero **adp\_to\_little-r.csh**

```
#cd ..  
#emacs adp_to_little-r.csh
```

realizando las siguientes modificaciones en negrita:

```
### DataDir MUST be the ABSOLUTE path to your data directory  
set DataDir = /home/mm5i/mm5v3/dadesobs/prova  
  
### INSFCData is both surface and ship data  
set InUpperData = ( $DataDir/RAOBS_A )  
set InSFCData = ( $DataDir/SFC6HR_A $DataDir/SFC3HR_A )  
  
set starting_date = 1993031300  
set ending_date = 1993031400  
  
# The amount of CPU time spent unpacking the entire  
# globe is embarrassing. As much as possible, restrict the  
# bounds.  
  
set lon_e = 180  
set lon_w = -180  
set lat_s = -90  
set lat_n = 90
```



```
set lon_e    = -40
set lon_w    = -130
set lat_s    = 20
set lat_n    = 70
```

para ejecutar el script de conversión, teclee:

```
./adp_to_little-r.csh >& log &
```

Edite el fichero **log** para comprobar si el proceso de ejecución fue el correcto, si es así al final del fichero debería aparecer un mensaje tal como:

Converting Upper-Air Data

```
~/mm5v3/FETCH
DONE with conversion
```

Los ficheros que se obtienen son del tipo:

**obs:aaaa-mm-dd\_hh** ficheros con los datos de las observaciones de los radiosondeos, para el año (aaaa), mes (mm) y día (dd) seleccionados y para las horas (hh) en la que hay datos

**surface\_obs\_r:aaaa-mm-dd\_hh** ficheros con los datos en superficie de estaciones meteorológicas, de barcos y boyas.

## 3.5 LITTLE\_R

Descompacte el fichero

**LITTLE\_R.TAR.gz**

en la carpeta

**/home/mm5i/mm5v3/LITTLE\_R**

Edite el fichero **Makefile** realizando los siguientes cambios en **negrita**

```
#emacs Makefile &
```



Manual de instalación del MM5 usando los compiladores  
FORTRAN de INTEL (IFC)

---

IFC:

```
I_HAVE_NCARG      =      all
I_DONT_HAVE_NCARG =      little_r
PROGS            =      $(I_HAVE_NCARG)
#PROGS            =      $(I_DONT_HAVE_NCARG)
INTEL_LIB        =      /opt/intel/compiler70/ia32/lib
```

...

```
if [ $$? = 0 ] ; then echo "Compiling for Linux" ; \
  echo "AR"          =      $(AR)"          > macros_little_r ; \
  echo "RM"          =      $(RM)"          >> macros_little_r ; \
  echo "RM_LIST"     =      $(RM_LIST)" >> macros_little_r ; \
  echo "CD"          =      $(CD)"          >> macros_little_r ; \
  echo "LN"          =      $(LN)"          >> macros_little_r ; \
  echo "MAKE"        =      $(MAKE)"        >> macros_little_r ; \
  echo "SHELL"       =      /bin/sh"        >> macros_little_r ; \
  echo "TOUCH"       =      touch"          >> macros_little_r ; \
  echo "CPP"         =      /lib/cpp"        >> macros_little_r ; \
  echo "CPPFLAGS"    =      -I. -C -P -DDEC -traditional" >>
  macros_little_r    ; \
  echo "FC"          =      ifc"          >> macros_little_r ; \
  echo "FCFLAGS"    =      -FR -pc32 " >> macros_little_r ; \
  echo "LDFLAGS"    =      -i_dynamic" >> macros_little_r ; \
  echo "CCFLAGS"     =      -DDEC -I." >> macros_little_r ; \
  echo "LOCAL_LIBRARIES =      -L$(NCARG_ROOT)/lib
-L/usr/X11R6/lib -lncarg -lncarg_gks -lncarg_c -lX11 -L$(INTEL_LIB)
-lPEPCF90 -L/usr/lib/gcc-lib/i386-redhat-linux/3.2.2 -lg2c" >>
  macros_little_r    ; \
  ( $(CD) src ; $(MAKE) $(PROGS) ) ; \
```



## Manual de instalación del MM5 usando los compiladores FORTRAN de INTEL (IFC)

En lugar de:

PGF:

```
I_HAVE_NCARG      =      all
I_DONT_HAVE_NCARG =      little_r
PROGS              =      $(I_HAVE_NCARG)
#PROGS            =      $(I_DONT_HAVE_NCARG)
...
```

```
else grep Linux .tmpfile ;\
if [ $$? = 0 ] ; then echo "Compiling for Linux" ;\
    echo "AR      =      $(AR)"      > macros_little_r ;\
    echo "RM      =      $(RM)"      >> macros_little_r ;\
    echo "RM_LIST =      $(RM_LIST)" >> macros_little_r ;\
    echo "CD      =      $(CD)"      >> macros_little_r ;\
    echo "LN      =      $(LN)"      >> macros_little_r ;\
    echo "MAKE    =      $(MAKE)"    >> macros_little_r ;\
    echo "SHELL   =      /bin/sh"    >> macros_little_r ;\
    echo "TOUCH   =      touch"      >> macros_little_r ;\
    echo "CPP     =      /lib/cpp"    >> macros_little_r ;\
    echo "CPPFLAGS =      -I. -C -P -DDEC -traditional" >>
macros_little_r ;\
    echo "FC      =      pgf90"      >> macros_little_r ;\
    echo "FCFLAGS =      -Mfreeform -pc 32 -byteswapio" >>
macros_little_r ;\
    echo "LDFLAGS =      "          >> macros_little_r ;\
    echo "CCFLAGS =      -DDEC -I." >> macros_little_r ;\
    echo "LOCAL_LIBRARIES =      -L$(NCARG_ROOT)/lib
-L/usr/X11R6/lib -lncarg -lncarg_gks -lncarg_c -lX11 -L$(PGI)/linux86/lib
-L/usr/lib -lf2c" >> macros_little_r ;\
    ( $(CD) src ; $(MAKE) $(PROGS) ) ;\
```

Para montar el módulo teclee:

```
#make >& make.out &
```

Edite el fichero **make.out** para comprobar que el proceso de montaje fue el correcto, si es así debería aparecer al final del fichero un mensaje tal como:

```
154 Lines Compiled
/bin/rm -f plot_soundings.f
ifc -o plot_soundings -FR -pc32 -i_dynamic plot_soundings.o module_mapinfo.o
module_report.o module_skewt.o -L/usr/local/ncarg/lib -L/usr/X11R6/lib -lncarg -
lncarg_gks -lncarg_c -lX11 -L/opt/intel/compiler70/ia32/lib -lPEPCF90 -L/usr/lib/gcc-
lib/i386-redhat-linux/3.2.2 -lg2c
make[1]: Saliendo directorio `/home/mm5i/mm5v3/LITTLE_R/src'
/bin/rm -f little_r ; ln -s src/little_r .
if [ all = all ] ; then echo "Building plot programs" ;\
    /bin/rm -f plot_level ; ln -s src/plot_level . ;\
    /bin/rm -f plot_soundings ; ln -s src/plot_soundings . ;\
```



## Manual de instalación del MM5 usando los compiladores FORTRAN de INTEL (IFC)

---

fi  
Building plot programs

Edite el fichero **namelist.input**, con los siguientes cambios:

```
#emacs namelist.input &
```

```
&record1
```

```
start_year      = 1993
start_month     = 03
start_day       = 13
start_hour      = 00
end_year        = 1993
end_month       = 03
end_day         = 14
end_hour        = 00
interval        = 43200/
```

```
&record2
```

```
fg_filename = './REGRID/regridder/REGRID_DOMAIN1'
obs_filename = '/home/mm5i/mm5v3/FETCH/obs:1993-03-13_00'
              '/home/mm5i/mm5v3/FETCH/obs:1993-03-13_06'
              '/home/mm5i/mm5v3/FETCH/obs:1993-03-13_12'
              '/home/mm5i/mm5v3/FETCH/obs:1993-03-13_18'
              '/home/mm5i/mm5v3/FETCH/obs:1993-03-14_00'
sfc_obs_filename = '/home/mm5i/mm5v3/FETCH/surface_obs_r:1993-03-13_00'
                  '/home/mm5i/mm5v3/FETCH/surface_obs_r:1993-03-13_03'
                  '/home/mm5i/mm5v3/FETCH/surface_obs_r:1993-03-13_06'
                  '/home/mm5i/mm5v3/FETCH/surface_obs_r:1993-03-13_09'
                  '/home/mm5i/mm5v3/FETCH/surface_obs_r:1993-03-13_12'
                  '/home/mm5i/mm5v3/FETCH/surface_obs_r:1993-03-13_15'
                  '/home/mm5i/mm5v3/FETCH/surface_obs_r:1993-03-13_18'
                  '/home/mm5i/mm5v3/FETCH/surface_obs_r:1993-03-13_21'
                  '/home/mm5i/mm5v3/FETCH/surface_obs_r:1993-03-14_00' /
```

Para ejecutar el módulo LITTLE\_R, teclee:

```
#!/little_r >& log &
```

Edite el fichero **log** para comprobar si el proceso de ejecución fue el correcto, si es así debería aparecer al final del fichero un mensaje tal como:

```
-----  
Time Loop Processing for SFC FDDA, date = 1993-03-14_00:00:00
```

```
STOP 99999
```



Los ficheros que se obtienen de la ejecución son:

**LITTLE\_R\_DOMAIN1**  
**SFCFDDA\_DOMAIN1**

### 3.6 INTERPF

Descompacte el fichero:

**INTERPF.TAR.gz**

en la carpeta:

**/home/mm5i/mm5v3/INTERPF**

Edite el fichero **Makefile**, realizando los siguientes cambios (en negrita):

IFC:

```
else grep Linux .tmpfile ; \
if [ $$? = 0 ] ; then echo "Compiling for Linux" ; \
    echo "AR" = $(AR)" > macros_interpf ; \
    echo "RM" = $(RM)" >> macros_interpf ; \
    echo "RM_LIST" = $(RM_LIST)" >> macros_interpf ; \
    echo "CD" = $(CD)" >> macros_interpf ; \
    echo "LN" = $(LN)" >> macros_interpf ; \
    echo "MAKE" = $(MAKE)" >> macros_interpf ; \
    echo "SHELL" = /bin/sh" >> macros_interpf ; \
    echo "TOUCH" = touch" >> macros_interpf ; \
    echo "CPP" = /lib/cpp" >> macros_interpf ; \
    echo "CPPFLAGS" = -I. -C -P -DDEC -traditional" >>
macros_interpf ; \
echo "FC" = ifc" >> macros_interpf ; \
echo "FCFLAGS" = -FR -pc32" >> macros_interpf ; \
echo "LDFLAGS" = -i_dynamic" >> macros_interpf ; \
    echo "CCFLAGS" = -DDEC -I." >> macros_interpf ; \
    echo "LOCAL_LIBRARIES" = " >> macros_interpf ; \
    ( $(CD) src ; $(MAKE) all ) ; \
```



## Manual de instalación del MM5 usando los compiladores FORTRAN de INTEL (IFC)

En lugar de:

PGF:

```
else grep Linux .tmpfile ; \
if [ $$? = 0 ] ; then echo "Compiling for Linux" ; \
echo "AR = $(AR)" > macros_interpf ; \
echo "RM = $(RM)" >> macros_interpf ; \
echo "RM_LIST = $(RM_LIST)" >> macros_interpf ; \
echo "CD = $(CD)" >> macros_interpf ; \
echo "LN = $(LN)" >> macros_interpf ; \
echo "MAKE = $(MAKE)" >> macros_interpf ; \
echo "SHELL = /bin/sh" >> macros_interpf ; \
echo "TOUCH = touch" >> macros_interpf ; \
echo "CPP = /lib/cpp" >> macros_interpf ; \
echo "CPPFLAGS = -I. -C -P -DDEC -traditional" >>
macros_interpf ; \
echo "FC = pgf90" >> macros_interpf ; \
echo "FCFLAGS = -Mfreeform -pc 32 -byteswapio" >>
macros_interpf ; \
echo "LDFLAGS = " >> macros_interpf ; \
echo "CCFLAGS = -DDEC -I." >> macros_interpf ; \
echo "LOCAL_LIBRARIES = " >> macros_interpf ; \
```

Para montar el módulo, teclee:

```
#make >& make.out&
```

Edite el fichero **make.out**, para comprobar que el proceso de montaje ha sido el correcto. Si todo fue bien debería aparecer al final del fichero un mensaje tal como:

```
1189 Lines Compiled
/bin/rm -f interpf.f
ifc -o interpf -i_dynamic interpf.o module_bdy.o module_date_pack.o module_diags.o
module_file.o module_header_data.o module_hydro_interp.o module_all_io.o
module_lateral_bdy.o module_nh_interp.o module_phys_consts.o module_util.o
ld: Aviso: el tamaño del símbolo `MODULE.all_io_1' cambió de 356 a 360 en
module_all_io.o
make[1]: Saliendo directorio `home/mm5v3/INTERPF/src'
( /bin/rm -f interpf ; ln -s src/interpf . )
```

donde home es la ruta del usuario que está usando el MM5.

Edite el fichero **namelist.input**, siguiendo las instrucciones del tutorial on-line del MM5. Si el paquete anterior usado fue el REGRID, en lugar del FETCH y el LITTLE\_R, debe especificarse la ruta del REGRID\_DOMAIN1:

```
/home/mm5i/mm5v3/REGRID/REGRID_DOMAIN1
```

si hace la prueba con el FETCH y LITTLE\_R, debería ser:



/home/mm5i/mm5v3/LITTLE\_R/LITTLE\_R\_DOMAIN1

Teclee lo siguiente para ejecutar el módulo INTERPF:

```
#!/interpf >& log &
```

Edite el fichero **log** para comprobar que no ha habido ningún error en el proceso de ejecución. Si todo fue bien, al final del fichero debería aparecer:

```
-----  
FINISHED INTERP FOR DOMAIN ID #1  
-----
```

```
STOP 99999
```

Se han de crear los ficheros

**LOWBDY\_DOMAIN1**  
**MMINPUT\_DOMAIN1**

## 3.7 MM5

Descompacte el fichero:

**MM5.TAR.gz**

en la carpeta:

**/home/mm5i/mm5v3/MM5**

Las siguientes explicaciones son sólo válidas si se utiliza el MM5 en un único ordenador, es decir si se utiliza el modelo en modo serie. No son válidas para correrlo en una granja de PC's, que sería en modo paralelo. Para ello véase la siguiente dirección:

- <http://acd.ufrj.br/~ricardom/mm5>

Edite el fichero **configure.user.linux**, con los siguientes cambios en negrita y guárdelo con el nombre de **configure.user**:





## Manual de instalación del MM5 usando los compiladores FORTRAN de INTEL (IFC)

---

IFC:

```
#-----  
# 3. Fortran options  
#-----  
.  
.  
.  
FC = ifc  
FCFLAGS = -I$(LIBINCLUDE) -O2 -tpp6  
#FCFLAGS = -I$(LIBINCLUDE) -O2 -Mcray=pointer -tp p6 -pc 32 -Mnoframe -  
byteswapio -mp \  
#-Mnosgimp  
CPP = /lib/cpp  
CFLAGS = -O  
CPPFLAGS = -I$(LIBINCLUDE)  
LDOPTIONS = -i_dynamic -Vaxlib -O2 -tpp6 -pc32  
#LDOPTIONS = -O2 -Mcray=pointer -tp p6 -pc 32 -Mnoframe -byteswapio -mp  
LOCAL_LIBRARIES = -L/opt/intel/compiler70/ia32/lib -lPEPCF90  
MAKE = make -i -r
```

En lugar de:

PGF:

```
#-----  
# 3. Fortran options  
#-----  
.  
.  
.  
FC = pgf90  
FCFLAGS = -I$(LIBINCLUDE) -O2 -Mcray=pointer -tp p6 -pc 32 -Mnoframe -  
byteswapio  
#FCFLAGS = -I$(LIBINCLUDE) -O2 -Mcray=pointer -tp p6 -pc 32 -Mnoframe -  
byteswapio -mp \  
#-Mnosgimp  
CPP = /lib/cpp  
  
CFLAGS = -O  
CPPFLAGS = -I$(LIBINCLUDE)  
LDOPTIONS = -O2 -Mcray=pointer -tp p6 -pc 32 -Mnoframe -byteswapio  
#LDOPTIONS = -O2 -Mcray=pointer -tp p6 -pc 32 -Mnoframe -byteswapio  
-mp  
LOCAL_LIBRARIES =  
MAKE = make -i -r
```

Siga las instrucciones que se dan en el tutorial on-line del MM5 para escoger las diferentes opciones para las parametrizaciones físicas de la simulación en un caso particular. Para probar el caso de prueba no hay que realizarse ninguna modificación en las parametrizaciones.



## Manual de instalación del MM5 usando los compiladores FORTRAN de INTEL (IFC)

---

El montaje de este módulo se realiza en dos partes. Para la primera parte teclee lo siguiente para montar el módulo:

```
#make >& make.out &
```

Edite el fichero **make.out** para comprobar si esta primera parte del proceso de montaje fue la correcta. Si es así, debería aparecer al final del fichero un mensaje del tipo:

```
671 Lines Compiled
rm -f mm5.exe
ifc -o mm5.exe mm5.o -i_dynamic -O2 -tpp6 -pc32 -L/opt/intel/compiler70/ia32/lib -
IPEPCF90 ../libutil.a
make[1]: Saliendo directorio `home/mm5v3/MM5/Run'
```

donde home es la ruta del usuario que está usando el MM5.

Para la segunda parte del proceso de montaje, teclee:

```
#make mm5.deck >& make.out.deck &
```

Edite el fichero **make.out.deck** para comprobar si la segunda parte del proceso de montaje fue el correcto. Si es así, debería aparecer al final del fichero un mensaje tal como:

```
Including file ./Templates/oparam
Including file ./Templates/lparam
Including file ./Templates/nparam
Including file ./Templates/pparam
Including file ./Templates/fparam
```

Ahora diríjase a la carpeta **/home/mm5i/mm5v3/MM5/Run** y genere los siguientes links o enlaces:

```
#ln -s ../INTERPF/MMINPUT_DOMAIN1
#ln -s ../INTERPF/BDYOUT_DOMAIN1
#ln -s ../INTERPF/LOWBDY_DOMAIN1
#ln -s ../TERRAIN/TERRAIN_DOMAIN2
```

Vuelva a la carpeta **MM5** y edite el fichero **mm5.deck**, siguiendo las instrucciones del tutorial on-line del MM5. En el caso de prueba (SOC) no se hacen modificaciones. Teclee lo siguiente para ejecutar el paquete:

```
#!/mm5.deck
```

El proceso de ejecución crea un fichero **log** en la carpeta **/home/mm5i/mm5v3/MM5/Run** llamado **mm5.print.out**, donde se encuentran escritos todos los pasos del proceso de ejecución. Edítelo para comprobar si todo fue correcto. Si es así, debería aparecer al final del fichero un mensaje tal como:

```
--- MODEL OUTPUT IS WRITTEN AT TIME = 720.00 MINUTES FOR DOMAIN 1
--- MODEL OUTPUT IS WRITTEN AT TIME = 720.00 MINUTES FOR DOMAIN 2
+++ REWINDING SAVE FILE FOR DOMAIN 1
```



## Manual de instalación del MM5 usando los compiladores FORTRAN de INTEL (IFC)

```
+++ RESTART FILE IS WRITTEN AT TIME = 720.00 MINUTES FOR DOMAIN
1. IXTIMR = 720
+++ REWINDING SAVE FILE FOR DOMAIN 2
+++ RESTART FILE IS WRITTEN AT TIME = 720.00 MINUTES FOR DOMAIN
2. IXTIMR = 720
FORTRAN STOP 99999
```

La duración del proceso depende de la máquina donde se ejecuta, el tipo de datos de la integración y de las parametrizaciones físicas definidas. En un equipo P-IV, 2.66 GHz, 512 MB DDR RAM, es aproximadamente de unos 6 minutos.

Se generan los ficheros:

```
MMOUT_DOMAIN1
MMOUT_DOMAIN2
SAVE_DOMAIN1
SAVE_DOMAIN2
```

### 3.8 INTERPB

Descompacte el fichero:

**INTERPB.TAR.gz**

en la carpeta:

**/home/mm5i/mm5v3/INTERPB**

Edite el fichero **Makefile**, realizando los siguientes cambios (en negrita):

IFC:

```
else grep Linux .tmpfile ; \
if [ $$? = 0 ] ; then echo "Compiling for Linux" ; \
echo "AR = $(AR)" > macros_interpb ; \
echo "RM = $(RM)" >> macros_interpb ; \
echo "RM_LIST = $(RM_LIST)" >> macros_interpb ; \
echo "CD = $(CD)" >> macros_interpb ; \
echo "LN = $(LN)" >> macros_interpb ; \
echo "MAKE = $(MAKE)" >> macros_interpb ; \
echo "SHELL = /bin/sh" >> macros_interpb ; \
echo "TOUCH = touch" >> macros_interpb ; \
echo "CPP = /lib/cpp" >> macros_interpb ; \
echo "CPPFLAGS = -I. -C -P -DDEC -traditional" >>
macros_interpb ; \
echo "FC = ifc" >> macros_interpb ; \
echo "FCFLAGS = -FR -pc32" >> macros_interpb ; \
echo "LDFLAGS = -i_dynamic" >> macros_interpb ; \
echo "CCFLAGS = -DDEC -I." >> macros_interpb ; \
echo "LOCAL_LIBRARIES = " >> macros_interpb ; \
( $(CD) src ; $(MAKE) all ) ; \
```



## Manual de instalación del MM5 usando los compiladores FORTRAN de INTEL (IFC)

En lugar de:

PGF:

```
else grep Linux .tmpfile ;\
if [ $$? = 0 ] ; then echo "Compiling for Linux" ;\
  echo "AR          = $(AR)" > macros_interpb ;\
  echo "RM          = $(RM)" >> macros_interpb ;\
  echo "RM_LIST     = $(RM_LIST)" >> macros_interpb ;\
  echo "CD          = $(CD)" >> macros_interpb ;\
  echo "LN          = $(LN)" >> macros_interpb ;\
  echo "MAKE        = $(MAKE)" >> macros_interpb ;\
  echo "SHELL       = /bin/sh" >> macros_interpb ;\
  echo "TOUCH       = touch" >> macros_interpb ;\
  echo "CPP         = /lib/cpp" >> macros_interpb ;\
  echo "CPPFLAGS    = -I. -C -P -DDEC -traditional">>
macros_interpb ;\
  echo "FC          = pgf90" >> macros_interpb ;\
  echo "FCFLAGS     = -Mfreeform -pc 32 -byteswapio" >>
macros_interpb ;\
  echo "LDFLAGS     = " >> macros_interpb ;\
  echo "CCFLAGS     = -DDEC -I." >> macros_interpb ;\
  echo "LOCAL_LIBRARIES = " >> macros_interpb ;\
  ( $(CD) src ; $(MAKE) all ) ;\
```

Para montar el módulo, teclee:

```
#make >& make.out &
```

Edite el fichero **make.out** para comprobar que el proceso de montaje ha sido el correcto. Si todo fue bien, debería aparecer al final del fichero un mensaje como:

```
732 Lines Compiled
/bin/rm -f interpb.f
ifc -o interpb -i_dynamic interpb.o module_all_io.o module_date_pack.o
module_diags.o module_header_data.o module_interp.o module_map_utils.o
module_phys_consts.o module_util.o module_xyll.o
ld: Aviso: el tamaño del símbolo `MODULE.all_io_1' cambió de 404 a 408 en
module_all_io.o
make[1]: Saliendo directorio `home/mm5v3/INTERPB/src'
( /bin/rm -f interpb ; ln -s src/interpb . )
```

donde home es la ruta del usuario que está usando el MM5.

Edite el fichero **namelist.input**, siguiendo las instrucciones del tutorial on-line del MM5. En el caso de prueba deben realizarse los siguientes cambios:



```
&record0
input_file = '../MM5/Run/MMOUT_DOMAIN1' /
&record1
start_year = 1993
start_month = 03
start_day = 13
start_hour = 00
end_year = 1993
end_month = 03
end_day = 13
end_hour = 12
interval = 21600 /
```

Para ejecutar el modulo teclee:

```
#!/interp >& log &
```

Edite el fichero **log** para comprobar que la ejecución ha ido bien. Si todo fue correcto, al final del fichero debería aparecer un mensaje tal como:

```
Completed all requested dates. Exiting program.
FORTRAN STOP Successful completion 99999
```

Este módulo debe ejecutarse para cada uno de los dominios, en el caso de prueba (SOC) en el que se han definido dos dominios se harán las siguientes modificaciones para el dominio 2 (en negrita):

```
&record0
input_file = '../MM5/Run/MMOUT_DOMAIN2' /
&record1
start_year = 1993
start_month = 03

start_day = 13
start_hour = 00
end_year = 1993
end_month = 03
end_day = 13
end_hour = 12
interval = 10800 /
```

```
#!/interp >& log &
```

Edite el fichero **log** para comprobar que la ejecución ha ido bien. Si todo fue correcto, al final del fichero debería aparecer un mensaje tal como:

```
Completed all requested dates. Exiting program.
FORTRAN STOP Successful completion 99999
```

Se obtienen los ficheros:



**MMOUTP\_DOMIN1**  
**MMOUTP\_DOMIN2**  
**REGRID\_DOMAIN1**  
**REGRID\_DOMAIN2**  
y ficheros del tipo **FILE\_MMOUTP.x** entre otros.

### 3.9 NESTDOWN

Descompacte el fichero:

**NESTDOWN.TAR.gz**

en la carpeta:

**/home/mm5i/mm5v3/NESTDOWN**

Dirijase a la carpeta anterior y edite el fichero **Makefile**, realizando los siguientes cambios (en negrita):

IFC:

```
else grep Linux .tmpfile ; \
if [ $$? = 0 ] ; then echo "Compiling for Linux" ; \
echo "AR" = $(AR)" > macros_nestdown ; \
echo "RM" = $(RM)" >> macros_nestdown ; \
echo "RM_LIST" = $(RM_LIST)" >> macros_nestdown ; \
echo "CD" = $(CD)" >> macros_nestdown ; \
echo "LN" = $(LN)" >> macros_nestdown ; \
echo "MAKE" = $(MAKE)" >> macros_nestdown ; \
echo "SHELL" = /bin/sh" >> macros_nestdown ; \
echo "TOUCH" = touch" >> macros_nestdown ; \
echo "CPP" = /lib/cpp" >> macros_nestdown ; \
echo "CPPFLAGS" = -I. -C -P -DDEC -traditional" >>
macros_nestdown ; \
echo "FC" = ifc" >> macros_nestdown ; \
echo "FCFLAGS" = -FR -pc32" >> macros_nestdown ; \
echo "LDFLAGS" = -i_dynamic" >> macros_nestdown ; \
echo "CCFLAGS" = -DDEC -I." >> macros_nestdown ; \
echo "LOCAL_LIBRARIES" = " >> macros_nestdown ; \
```



## Manual de instalación del MM5 usando los compiladores FORTRAN de INTEL (IFC)

En lugar de:

PGF:

```
else grep Linux .tmpfile ; \
if [ $$? = 0 ] ; then echo "Compiling for Linux" ; \
echo "AR = $(AR)" > macros_nestdown ; \
echo "RM = $(RM)" >> macros_nestdown ; \
echo "RM_LIST = $(RM_LIST)" >> macros_nestdown ; \
echo "CD = $(CD)" >> macros_nestdown ; \
echo "LN = $(LN)" >> macros_nestdown ; \
echo "MAKE = $(MAKE)" >> macros_nestdown ; \
echo "SHELL = /bin/sh" >> macros_nestdown ; \
echo "TOUCH = touch" >> macros_nestdown ; \
echo "CPP = /lib/cpp" >> macros_nestdown ; \
echo "CPPFLAGS = -I. -C -P -DDEC -traditional">>
macros_nestdown ; \
echo "FC = pgf90" >> macros_nestdown ; \
echo "FCFLAGS = -Mfreeform -pc 32 -byteswapio" >>
macros_nestdown ; \
echo "LDFLAGS = " >> macros_nestdown ; \
echo "CCFLAGS = -DDEC -I." >> macros_nestdown ; \
echo "LOCAL_LIBRARIES = " >> macros_nestdown ; \
```

Para montar el módulo, teclee:

```
#make >& make.out &
```

Edite el fichero **make.out** para comprobar si el proceso de montaje fue el correcto. Si es así, debería aparecer al final del fichero un mensaje tal como:

```
1833 Lines Compiled
/bin/rm -f nestdown.f
ifc -o nestdown -i_dynamic nestdown.o module_base_state.o module_bdy.o
module_date_pack.o module_file.o module_header_data.o module_horiz_interp.o
module_lateral_bdy.o module_all_io.o module_vert_interp.o module_util.o
ld: Aviso: el tamaño del símbolo `MODULE.all_io_1' cambió de 476 a 480 en
module_all_io.o
make[1]: Saliendo directorio `home/mm5v3/NESTDOWN/src'
( /bin/rm -f nestdown ; ln -s src/nestdown . )
```

donde home es la ruta del usuario que está usando el MM5.

El módulo nestdown tiene dos maneras de proceder. En estas notas sólo se presenta una de ellas, en la que se usa solamente la salida de una simulación para el dominio madre y los datos del terreno para el dominio anidado (**Figura 8**).

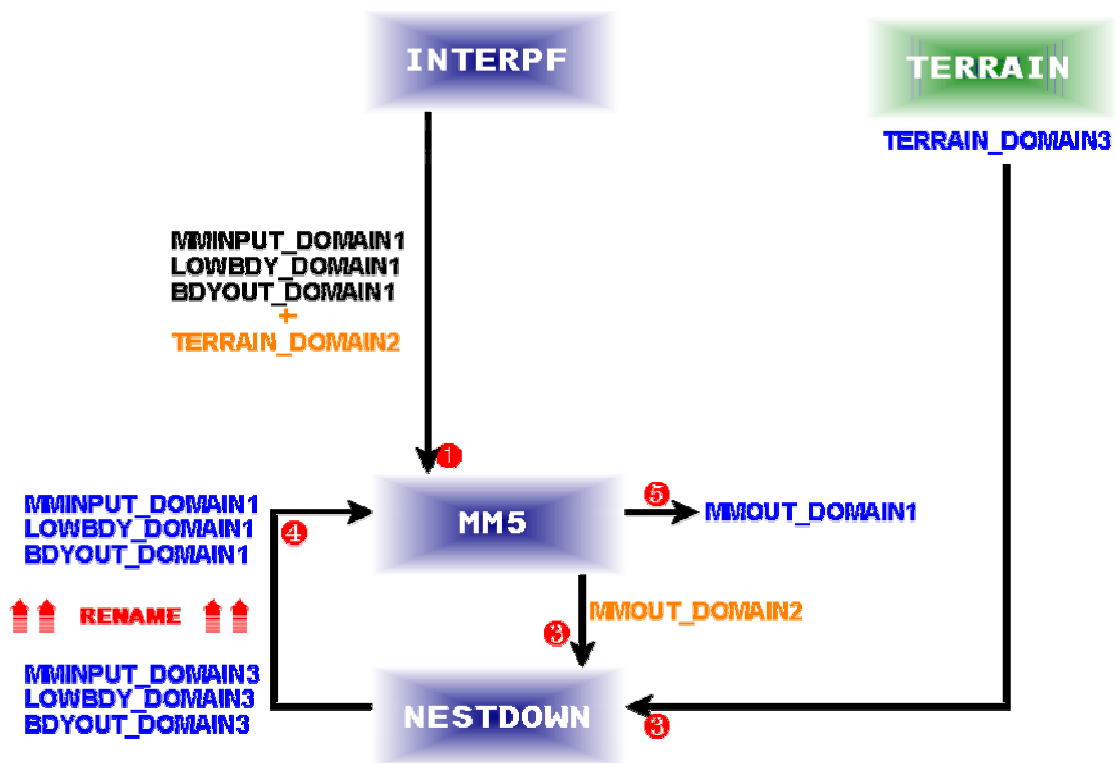


Figura 8. Esquema de anidamiento a través del NESTDOWN

Esta opción requiere que se vuelva a ejecutar el módulo TERRAIN incorporando la información de un tercer dominio con una resolución de 10 km para el caso de prueba, por lo tanto hay que irse al modulo TERRAIN y limpiarlo.

```
#cd ..
#cd TERRAIN
#make clean
```

No borre el **terrain.deck**, edítelo y añada lo siguiente (en negrita) para definir el tercer dominio:

```
#emacs terrain.deck
```

```
...
MAXNES = 3, ; NUMBER OF DOMAINS TO PROCESS
NESTIX = 35, 49, 31, 181, 211, 221, ; GRID DIMENSIONS IN Y DIRECTION
NESTJX = 41, 52, 31, 196, 211, 221, ; GRID DIMENSIONS IN X DIRECTION
DIS = 90., 30., 10., 3.0, 1.0, 1.0, ; GRID DISTANCE
NUMNC = 1, 1, 2, 3, 4, 5, ; MOTHER DOMAIN ID
NESTI = 1, 10, 18, 35, 45, 50, ; LOWER LEFT I OF NEST IN MOTHER
DOMAIN
NESTJ = 1, 17, 19, 65, 55, 50, ; LOWER LEFT J OF NEST IN MOTHER
DOMAIN
RID = 1.5, 1.5, 1.5, 3.1, 2.3, 2.3, ; RADIUS OF INFLUENCE IN GRID UNITS
(IFANAL=T)
NTYPE = 2, 3, 4, 6, 6, 6, ; INPUT DATA RESOLUTION
```





## Manual de instalación del MM5 usando los compiladores FORTRAN de INTEL (IFC)

---

```
;  
; 1: 1 deg (~111 km) global terrain and landuse  
; 2: 30 min ( ~56 km) global terrain and landuse  
; 3: 10 min ( ~19 km) global terrain and landuse  
; 4; 5 min ( ~9 km) global terrain and landuse  
; 5; 2 min ( ~4 km) global terrain and landuse  
; 6; 30 sec ( ~.9 km) global terrain and landuse  
;  
NSTTYP= 1, 2, 1, 2, 2, 2, ; 1 -- ONE WAY NEST, 2 -- TWO WAY NEST  
...
```

ejecute de nuevo el módulo TERRAIN

```
#!/terrain.deck
```

obteniendo los ficheros:

```
TERRAIN_DOMIAN1  
TERRAIN_DOMAIN2  
TERRAIN_DOMAIN3  
TER.PLT
```

Vuelva al directorio NESTDOWN y edite el fichero **namelist.input** con lo siguiente:

```
#cd ..  
#cd NESTDOWN  
#emacs namelist.input  
  
&record0  
input_file = '../MM5/Run/MMOUT_DOMAIN2'  
input_lowbdy_file = '../MM5/Run/LOWBDY_DOMAIN2'  
input_terrain_file = '../TERRAIN/TERRAIN_DOMAIN3' /  
  
&record1  
start_year = 1993  
start_month = 03  
start_day = 13  
start_hour = 03  
end_year = 1993  
end_month = 03  
end_day = 13  
end_hour = 06  
interval = 3600  
less_than_24h = .TRUE. /  
  
&record2  
sst_to_ice_threshold = -9999 /  
  
&record4  
wrth2o = .TRUE. /
```



## Manual de instalación del MM5 usando los compiladores FORTRAN de INTEL (IFC)

---

```
&record5  
ifdatim      = 1 /
```

```
&record6  
interp_method = 1  
use_mm5_lowbdy = .FALSE. /
```

Si se va a correr el módulo para una integración superior a 24 horas la variable `less_than_24h` tiene que ser igual a `.FALSE.`

Para ejecutar el módulo NESTDOWN, teclee:

```
#!/nestdown >& log &
```

Edite el fichero **log** para comprobar si la ejecución del paquete fue la correcta, si es así debería aparecer al final del fichero un mensaje del tipo:

```
Lateral boundary conditions valid from 1993-03-13_05:00:00 through 1993-03-  
13_06:00:00.  
At end of requested time periods.  
STOP 99999
```

Los ficheros que se obtienen de este proceso son

```
MMINPUT_DOMAIN3  
BDYOUT_DOMAIN3  
LOWBDY_DOMAIN3
```

Estos ficheros se usarán para correr el anidamiento unidireccional (one-way nesting) del tercer dominio en el paquete MM5. Hay que dirigirse a la carpeta **/home/mm5i/mm5v3/MM5/Run:**

```
# cd ..  
# cd MM5
```

```
# cd Run
```

eliminar todos los links y ficheros de salida de la integración anterior si los hubiera:

```
#rm MMINPUT_DOMAIN1  
#rm LOWBDY_DOMAIN1  
#rm BDYOUT_DOMAIN1  
#rm TERRAIN_DOMAIN2  
#rm MMOUT_DOMAIN1  
#rm MMOUT_DOMAIN2
```

Hay que copiar los ficheros de salida obtenidos al ejecutar NESTDOWN y renombrarlos como de dominio 1:

```
#cp /home/mm5i/mm5v3/NESTDOWN/MMINPUT_DOMAIN3 MMINPUT_DOMAIN1
```



## Manual de instalación del MM5 usando los compiladores FORTRAN de INTEL (IFC)

---

```
#cp /home/mm5i/mm5v3/NESTDOWN/BDYOUT_DOMAIN3 BDYOUT_DOMAIN1  
#cp /home/mm5i/mm5v3/NESTDOWN/LOWBDY_DOMAIN3 LOWBDY_DOMAIN1
```

subir al directorio MM5

```
#cd ..
```

limpiar el directorio

```
#make clean
```

editar el **configure.user**, donde se especificará las parametrizaciones físicas para el tercer dominio así como el hecho de contar ahora con una integración para un solo dominio (cambios a realizar en negrita):

```
#emacs configure.user
```

```
# Sections  
# This configure.user.linux file is used to compile on PC running linux only.  
# For options to compile on Unix systems, please use configure.user  
# 1. System Variables  
# 2. User Variables  
# 3. Fortran options  
# 3i. PC_PGF77 (Linux)  
# 4. General commands  
# 5. Options for making "./include/parame.incl"  
# 6. Physics Options (memory related)  
#  
#-----  
# 1. System Variables  
#-----  
SHELL = /bin/sh  
.SUFFIXES: .F .i .o .f  
#-----  
# 2. User Variables  
#-----  
# RUNTIME_SYSTEM - Currently supported systems.  
# PC_PGF77  
RUNTIME_SYSTEM = "PC_PGF77"  
#  
#-----  
# 3. Fortran options  
#-----  
LIBINCLUDE = $(DEVTOP)/include  
#-----  
# 3i. PC running Linux and using pgf77 compiler  
# You may also need to unlimit stacksize by typing  
# limit stacksize unlimited  
# or  
# setenv MPSTKZ 8M
```



## Manual de instalación del MM5 usando los compiladores FORTRAN de INTEL (IFC)

---

```
# (or bigger.)
#
# If your compiler supports both SGI and OpenMP parallel directives,
# you need -Mnosgimp in your FCFLAGS. Otherwise remove it.
#
# Add -Kieeee option if you compile Gayno-Seaman PBL scheme (IBLTYP=6).
# MM5 will fail without this option until the compiler bug is fixed.
#-----
# 3i1. PC_PGF77 (LINUX/Portland Group Inc.)
# pgf77 version 1.6 and above
# May use pgf90 if the version is 3.1-4
#-----
FC = ifc
FCFLAGS = -I$(LIBINCLUDE) -O2 -tpp6
#FCFLAGS = -I$(LIBINCLUDE) -O2 -Mcray=pointer -tp p6 -pc 32 -Mnoframe -
byteswapio -mp \
#-Mnosgimp
CPP = /lib/cpp
CFLAGS = -O
CPPFLAGS = -I$(LIBINCLUDE)
LDOPTIONS = -i_dynamic -O2 -tpp6 -pc32
#LDOPTIONS = -O2 -Mcray=pointer -tp p6 -pc 32 -Mnoframe -byteswapio -mp
LOCAL_LIBRARIES = -L/opt/intel/compiler70/ia32/lib -IPEPCF90
MAKE = make -i -r
#-----
# 4. General commands
#-----
AR = ar ru
RM = rm -f
RM_CMD = $(RM) *.CKP *.ln *.BAK *.bak *.o *.i core errs ,* *~ *.a \
.emacs_* tags TAGS make.log MakeOut *.f !
GREP = grep -s
CC = cc
#-----
# 5. Options for making ./include/parame.incl
#-----
#
# FDDAGD (integer) - "1" -> FDDA gridded run
FDDAGD = 0
#
# FDDAOBS (integer) - "1" -> FDDA obs run
FDDAOBS = 0
#
# MAXNES (integer) - Max Number of Domains in simulation
MAXNES = 1
#
# MIX,MJX (integer) - Maximum Dimensions of any Domain
MIX = 49
```



Manual de instalación del MM5 usando los compiladores  
FORTRAN de INTEL (IFC)

---

```
MJX = 64
# MKX (integer)          - Number of half sigma levels in model
MKX = 23
#-----
# 6. Physics Options
# The first MAXNES values in the list will be used for the corresponding
# model nests; the rest in the list can be used to compile other options.
# The exception is FRAD, of which only the first value is used in the model,
# (i.e., only one radiation option is used for all nests). The rest allow
# other options to be compiled.
# Compilation of Arakawa-Schubert cumulus scheme requires imsl.
#-----
# IMPHYS - for explicit moisture schemes (array,integer)
IMPHYS = "4,1,1,1,1,1,1,1,1"
# - Dry,stable,warm rain,simple ice,mix phase,
# - 1 ,2 ,3 ,4 ,5
# - graupel(gsf),graupel(reisner2),schultz
# -,6 ,7 ,8
MPHYSTBL = 0
# - 0=do not use look-up tables for moist
# physics
# - 1=use look-up tables for moist physics
# (currently only simple ice and mix phase
# are available)
#
# ICUPA - for cumulus schemes (array,integer)
# - None,Kuo,Grell,AS,FC,KF,BM,KF2 - 1,2,3,4,5,6,7,8
ICUPA = "1,1,1,1,1,1,1,1,1"
#
# IBLTYP - for planetary boundary layer (array,integer)
# - 0=no PBL fluxes,1=bulk,2=Blackadar,
# 3=Burk-Thompson,4=Eta M-Y,5=MRF,
# 6=Gayno-Seaman,7=Pleim-Xiu
IBLTYP = "5,2,2,2,2,2,2,2,1"
#
# FRAD - for atmospheric radiation (integer)
# - Radiation cooling of atmosphere
# 0=none,1=simple,2=cloud,3=ccm2,rrtm=4
FRAD = "0,0,0,0,0"
#
# IPOLAR - (integer) for polar model used only if ISOIL=1
# 0=not polar (5-layer soil model)
# 1=polar (7-layer snow/soil model)
IPOLAR = 0
#
# ISOIL - for multi-layer soil temperature model (integer)
# - 0=no,1=yes (only works with IBLTYP=2,5)
# 2=Noah land-surface scheme (IBLTYP=4,5 only)
```



# Manual de instalación del MM5 usando los compiladores FORTRAN de INTEL (IFC)

```

#           3=Pleim-Xiu LSM (IBLTYP=7 only)
ISOIL = 1
#
# ISHALLO (array,integer)      - Shallow Convection Option
#           1=shallow convection,0=No shallow convection
ISHALLO = "0,0,0,0,0,0,0,0,0,0"
#-----
# Don't touch anything below this line
#-----
.c.o:
    $(RM) $@ && \
    $(CC) -c $(CFLAGS) $*.c
.F.o:
    $(RM) $@
    $(FC) -c $(FCFLAGS) $*.F
.f.o:
    $(RM) $@
    $(FC) -c $(FCFLAGS) $*.f

```

monte el módulo:

```
#make
```

cree el **mm5.deck** para esta nueva integración

```
#make mm5.deck
```

edite el fichero **mm5.deck** para que conste que la integración es ahora para un solo dominio (cambios en negrita):

....

```

cat > ./Run/oparam << EOF
&OPARAM
;
; ***** FORECAST TIME AND TIME STEP *****
;
;
TIMAX = 180.,      ; forecast length in minutes
TISTEP = 93.,     ; coarse domain DT in model, use 3*DX
;
; ***** OUTPUT/RESTART OPTIONS *****
;
;
IFREST = .FALSE.,   ; whether this is a restart
IXTIMR = 180,     ; restart time in minutes
IFSAVE = .TRUE.,    ; save data for restart
SVLAST = .TRUE.,   ; T: only save the last file for restart
                    ; F: save multiple files
SAVFRQ = 360.,     ; how frequently to save data (in minutes)

```









A partir de aquí se sigue como en el caso habitual, el siguiente módulo sería el INTERPB.

### 3.10 PAQUETES GRÁFICOS

Una vez se tienen los ficheros con la salida del modelo, se deben visualizar, para ello se pueden utilizar varias opciones, dependiendo de las necesidades, y hábitos de cada usuario. Vamos a exponer unas cuantas opciones de las más utilizadas, aunque no las únicas.

- GrADS
- Surfer
- Vis5D
- NCAR Graphics
- RIP
- Otros formatos

Paso intermedio Vis5D

vrml

gif

post script

post script color

24 ppm

xml

A modo de resumen véase la **Figura 9**

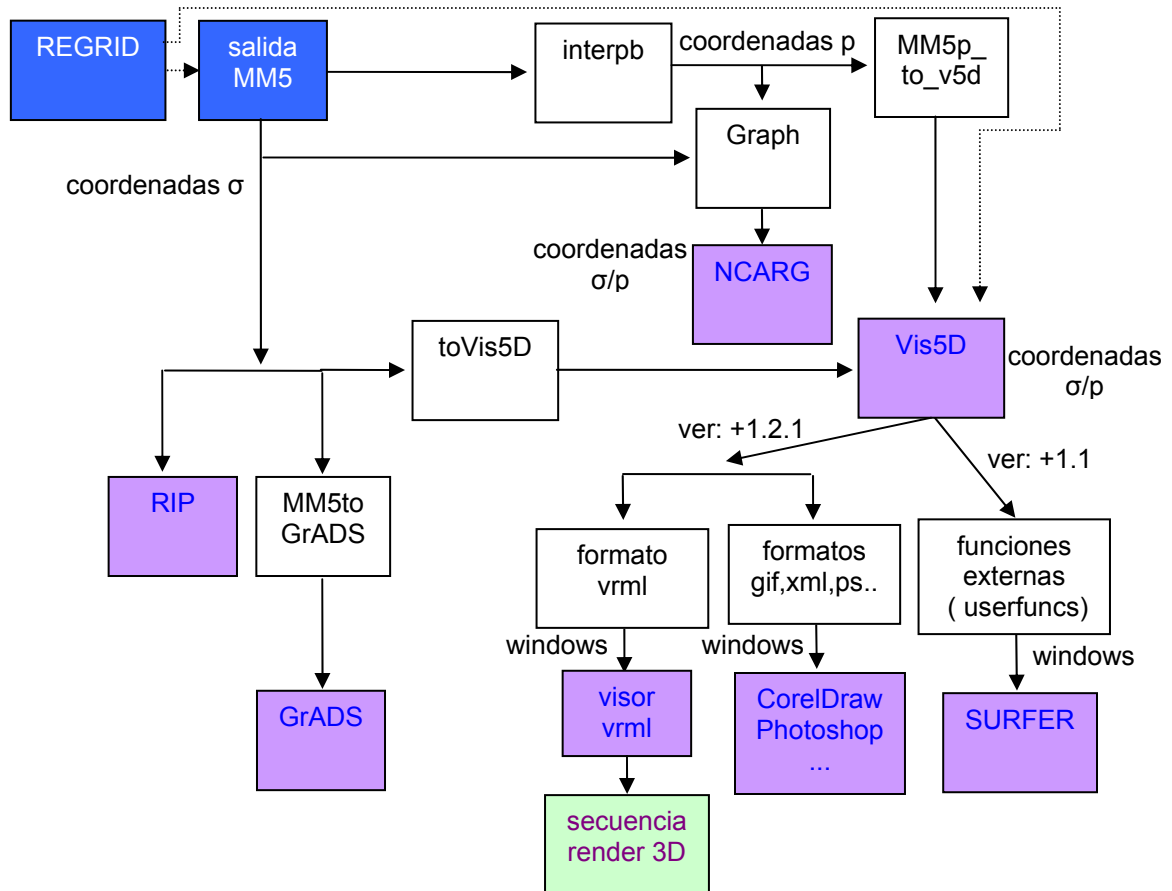


Figura 9. Esquema de las diferentes opciones de visualización de las salidas del modelo.

### 3.10.1 Conversión al VIS5D+

- **MM5p\_to\_v5d**

Script propio del grupo de Meteorología de la *Universitat de les Illes Balears*, creado por el Dr. Romualdo Romero March, para convertir la salida del modelo en coordenadas p al Vis5D+. Este script está disponible libremente en la dirección:

- <http://redibericamm5.uib.es/algoritmos.htm>

Los ficheros a descargar son:

**MM5p\_to\_v5d.f**  
**MM5p\_to\_v5d.f.m**  
**pars.MM5p**

Cópielos en la carpeta:

**/home/mm5i/mm5v3/PASOVIS5D**



## Manual de instalación del MM5 usando los compiladores FORTRAN de INTEL (IFC)

Este script está pensado para usar el compilador FORTRAN de GNU, que ya está incluido en el Red Hat 9. Para usarlo con el compilador FORTRAN de INTEL, debe realizar lo siguiente:

Edite el fichero **MM5p\_to\_v5d.f.m** y realice los siguientes cambios (en negrita):

IFC:

```
# If DEC or Linux, (Little-endian), add -DLITTLE to CFLAGS
```

```
# If AIX, remove the -DUNDERScore
```

```
CFLAGS = -c -g -DUNDERScore -DLITTLE
```

```
FFLAGS = -c -g
```

```
CC = cc
```

```
F77 = ifc
```

```
LIBS = -lm -i_dynamic -Vaxlib
```

En lugar de

GNU:

```
# If DEC or Linux, (Little-endian), add -DLITTLE to CFLAGS
```

```
# If AIX, remove the -DUNDERScore
```

```
CFLAGS = -c -g -DUNDERScore
```

```
FFLAGS = -c -g
```

```
CC = cc
```

```
F77 = f77
```

```
LIBS = -lm
```

Para montar el paquete teclee:

```
#make -f MM5p_to_v5d.f.m >& make.out &
```

Edite el fichero **make.out** para comprobar si el proceso de montaje fue el correcto. Si todo fue bien, debería aparecer en el fichero un mensaje tal como:

```
ifc -c -g -I/usr/local/vis5d/src MM5p_to_v5d.f
cc -c -g -DUNDERScore -DLITTLE /usr/local/vis5d/src/binio.c -o binio.o
cc -c -g -DUNDERScore -DLITTLE /usr/local/vis5d/src/v5d.c -o v5d.o
ifc MM5p_to_v5d.o binio.o v5d.o -lm -i_dynamic -Vaxlib -o MM5p_to_v5d
```

Edite el fichero **pars.MM5p** según las instrucciones del documento descriptivo **README.pdf** que se encuentra en la dirección:

- <http://redibericamm5.uib.es/algoritmos.htm>

En el caso de prueba (SOC), se pondrá los mismos datos de fechas y horas que los del fichero INTERPB, los cambios son:

```
##### USER CHOICES TO PRODUCE V5D file from pressure-surfaces MM5p output
#
*** MM5p output file
'/home/mm5i/mm5v3/INTERPB/MMOUTP_DOMAIN1'
#
```



Manual de instalación del MM5 usando los compiladores  
FORTRAN de INTEL (IFC)

---

```
*** Requested BEGIN date in format YYYY-MM-DD_hh:mm:ss
1993-03-13_00:00:00
*** Requested END date in format YYYY-MM-DD_hh:mm:ss
1993_03-13_12:00:00
*** Requested Time INTERVAL (in seconds)
21600
#
*** Compressmode for V5D data (1, 2 or 4 bytes per grid point)
4
#
*** Time-mean fields Tm, Um, Vm, Hm and RHm if selected (0:NO, 1:YES) ?
0
*** BEGIN, END and INTERVAL to compute time-mean fields
1993-03-13_00:00:00
1993-03-13_12:00:00
21600
#
```

Para ejecutar el script, teclee:

```
#./MM5p_to_v5d pars.MM5p nombre_fichero1.v5d >& log &
```

Edite el fichero **log** para comprobar si la ejecución del paquete fue la correcta, si es así debería aparecer al final del fichero un mensaje del tipo:

```
... SUCCESSFUL PROCESS OF ALL REQUESTED DATES !!!
V5D CLOSE OK
```

Obtendremos el fichero ***nombre\_fichero1.v5d***

Ahora repita la operación para el segundo dominio haciendo los siguientes cambios en el fichero **pars.MM5p**:

```
##### USER CHOICES TO PRODUCE V5D file from pressure-surfaces MM5p output
#
*** MM5p output file
'/home/mm5i/mm5v3/INTERPB/MMOUTP_DOMAIN2'
#
*** Requested BEGIN date in format YYYY-MM-DD_hh:mm:ss
1993-03-13_00:00:00
*** Requested END date in format YYYY-MM-DD_hh:mm:ss
1993_03-13_12:00:00
*** Requested Time INTERVAL (in seconds)
10800
#
*** Compressmode for V5D data (1, 2 or 4 bytes per grid point)
4
#
*** Time-mean fields Tm, Um, Vm, Hm and RHm if selected (0:NO, 1:YES) ?
```



0

\*\*\* BEGIN, END and INTERVAL to compute time-mean fields

**1993-03-13\_00:00:00**

**1993-03-13\_12:00:00**

**10800**

#

Para ejecutar el script, teclee:

```
#./MM5p_to_v5d pars.MM5p nombre_fichero2.v5d >& log &
```

Edite el fichero **log** para comprobar si la ejecución del paquete fue la correcta, si es así debería aparecer al final del fichero un mensaje del tipo:

```
... SUCCESSFUL PROCESS OF ALL REQUESTED DATES !!!  
V5D CLOSE OK
```

Obtendremos el fichero ***nombre\_fichero2.v5d***

Para visualizar los ficheros obtenidos mediante el proceso anterior, copie estos ficheros en una carpeta nueva carpeta dentro de la carpeta **mm5v3**, de nombre **prueba**, por ejemplo. Una vez copiados ahí los ficheros para visualizarlos haga:

Para el dominio 1:

```
#vis5d nombre_fichero1.v5d -mbs 0 -date
```

y para el dominio 2:

```
#vis5d nombre_fichero2.v5d -mbs 0 -date
```

para verlos simultáneamente:

```
#vis5d nombre_fichero1.v5d -date nombre_fichero2.v5d -mbs 0 -date
```

- **REGRID\_to\_v5d**

Script propio del grupo de Meteorología de la *Universitat de les Illes Balears*, creado por el Dr. Romualdo Romero March, para convertir las salidas del módulo REGRID (**REGRID\_DOMAINX**) al Vis5D+. Este script está disponible libremente en la dirección:

- <http://redibericamm5.uib.es/algoritmos.htm>

Los ficheros a descargar son:

```
REGRID_to_v5d.f  
REGRID_to_v5d.f.m  
pars.REGRID
```

Cópielos en la carpeta:



## Manual de instalación del MM5 usando los compiladores FORTRAN de INTEL (IFC)

### **/home/mm5i/mm5v3/PASOVIS5D**

Este script está pensado para usar el compilador FORTRAN de GNU, que ya está incluido en el Red Hat 9. Para usarlo con el compilador FORTRAN de INTEL, debe realizar lo siguiente:

Edite el fichero **REGRID\_to\_v5d.f.m** y realice los siguientes cambios (en negrita):

IFC:

# If DEC or Linux, (Little-endian), add -DLITTLE to CFLAGS

# If AIX, remove the -DUNDERScore

CFLAGS = -c -g -DUNDERScore -DLITTLE

FFLAGS = -c -g

CC = cc

**F77 = ifc**

**LIBS = -lm -i\_dynamic -Vaxlib**

En lugar de

GNU:

# If DEC or Linux, (Little-endian), add -DLITTLE to CFLAGS

# If AIX, remove the -DUNDERScore

CFLAGS = -c -g -DUNDERScore

FFLAGS = -c -g

CC = cc

**F77 = f77**

**LIBS = -lm**

Para montar el paquete teclee:

```
#make -f REGRID_to_v5d.f.m >& make.out &
```

Edite el fichero **make.out** para comprobar si el proceso de montaje fue el correcto. Si todo fue bien, debería aparecer en el fichero un mensaje tal como:

```
ifc -c -g -I/usr/local/vis5d/src REGRID_to_v5d.f
cc -c -g -DUNDERScore -DLITTLE /usr/local/vis5d/src/binio.c -o binio.o
cc -c -g -DUNDERScore -DLITTLE /usr/local/vis5d/src/v5d.c -o v5d.o
ifc REGRID_to_v5d.o binio.o v5d.o -lm -i_dynamic -Vaxlib -o REGRID_to_v5d
```

Edite el fichero **pars.REGRID** según las instrucciones del documento descriptivo **README.pdf** que se encuentra en la dirección:

- <http://redibericamm5.uib.es/algoritmos.htm>

En el caso de prueba (SOC), se pondrá los mismos datos de fechas y horas que los que aparecen en el namelist del regrider. Los cambios son:

```
##### USER CHOICES TO PRODUCE V5D file from pressure-surfaces REGRID
output
#
```



## Manual de instalación del MM5 usando los compiladores FORTRAN de INTEL (IFC)

---

```
*** REGRID output file
'/home/mm5i/mm5v3/REGRID/regridder/REGRID_DOMAIN1'
#
*** Requested BEGIN date in format YYYY-MM-DD_hh:mm:ss
1993-03-13_00:00:00
*** Requested END date in format YYYY-MM-DD_hh:mm:ss
1993_03-14_00:00:00
*** Requested Time INTERVAL (in seconds)
21600
#
*** Compressmode for V5D data (1, 2 or 4 bytes per grid point)
4
#
```

Para ejecutar el script, teclee:

```
#!/REGRID_to_v5d pars.REGRID nombre_fichero1.v5d >& log &
```

Edite el fichero **log** para comprobar si la ejecución del paquete fue la correcta, si es así debería aparecer al final del fichero un mensaje del tipo:

```
... SUCCESSFUL PROCESS OF ALL REQUESTED DATES !!!
V5D CLOSE OK
```

Obtendremos el fichero ***nombre\_fichero1.v5d***

Ahora repita la operación para el segundo dominio haciendo los siguientes cambios en el fichero **pars.REGRID**:

```
##### USER CHOICES TO PRODUCE V5D file from pressure-surfaces MM5p output
#
*** REGRID output file
'/home/mm5i/mm5v3/REGRID/REGRID_DOMAIN2'
#
*** Requested BEGIN date in format YYYY-MM-DD_hh:mm:ss
1993-03-13_00:00:00
*** Requested END date in format YYYY-MM-DD_hh:mm:ss
1993_03-14_00:00:00
*** Requested Time INTERVAL (in seconds)
21600
#
*** Compressmode for V5D data (1, 2 or 4 bytes per grid point)
4
#
```

Para ejecutar el script, teclee:

```
#!/REGRID_to_v5d pars.REGRID nombre_fichero2.v5d >& log &
```



## Manual de instalación del MM5 usando los compiladores FORTRAN de INTEL (IFC)

---

Edite el fichero **log** para comprobar si la ejecución del paquete fue la correcta, si es así debería aparecer al final del fichero un mensaje del tipo:

```
... SUCCESSFUL PROCESS OF ALL REQUESTED DATES !!!  
V5D CLOSE OK
```

Obtendremos el fichero **nombre\_fichero2.v5d**

Para visualizar los ficheros obtenidos mediante el proceso anterior, copie estos ficheros en una carpeta nueva carpeta dentro de la carpeta **mm5v3**, de nombre **prueba**, por ejemplo. Una vez copiados ahí los ficheros para visualizarlos haga:

Para el dominio 1:

```
#vis5d nombre_fichero1.v5d -mbs 0 -date
```

y para el dominio 2:

```
#vis5d nombre_fichero2.v5d -mbs 0 -date
```

para verlos simultáneamente:

```
#vis5d nombre_fichero1.v5d -date nombre_fichero2.v5d -mbs 0 -date
```

### • TOVIS5D

Este módulo del modelo MM5 convierte la salida del modelo en coordenadas sigma a formato Vis5D.

Descompacte el fichero:

```
TOVIS5D.TAR.gz
```

en la carpeta:

```
/home/mm5i/mm5v3/TOVIS5D
```

Diríjase a la carpeta anterior y edite el fichero **Makefile** con los siguientes cambios a realizar en negrita:

```
# cd /home/mm5i/mm5v3/TOVIS5D  
# emacs Makefile &
```





Manual de instalación del MM5 usando los compiladores  
FORTRAN de INTEL (IFC)

---

IFC:

linux:

```
cd src/ ; $(MAKE) target \
"FC = ifc" \
"FCFLAGS = -FR -DLINUX -I. " \
"CCFLAGS = -g -DLITTLE -DUNDERSCORE -c" \
"LIBS = -Vaxlib"
$(RM) tovis5d ; $(LN) src/tovis5d .
```

En lugar de:

PGF:

linux:

```
cd src/ ; $(MAKE) target \
"FC = pgf90" \
"FCFLAGS = -Mfreeform -DLINUX -I. -byteswapio " \
"CCFLAGS = -g -DLITTLE -DUNDERSCORE -c" \
"LIBS = "
$(RM) tovis5d ; $(LN) src/tovis5d .
```

Edite el fichero **tovis5d.csh** cambiando lo siguiente (en negrita):

```
#emacs tovis5d.csh &
```

IFC:

```
#
if ( ! -e $1 ) then
    echo "The file $1 does not exist"
    exit 1
endif
#
```

```
./tovis5d $1 >&! tovis5d.log
```

En lugar de:

PGF:

```
#
if ( ! -e $1 ) then
    echo "The file $1 does not exist"
    exit 1
endif
#
```

```
tovis5d $1 >&! tovis5d.log
```

Teclee lo siguiente para montar el módulo:

```
make linux >& make.out &
```



## Manual de instalación del MM5 usando los compiladores FORTRAN de INTEL (IFC)

Edite el fichero **make.out** para comprobar que todo fue correcto. Si es así, debería aparecer al final del fichero un mensaje tal como:

```
(ifc -i_dynamic -o tovis5d module_parameter.o module_variables.o advance_cymdh.o  
allocate_array.o bvfricalc.o ceilingcalc.o close_file.o cross_2_dot.o cttcalc.o ddpcalc.o  
deal_new_var.o decouple.o default_param.o derivcz.o diff_hour.o eqthecalc.o esat.o  
extingcalc.o fields.o fregcalc.o generate_difference.o get_date_time.o get_v2_param.o  
get_v3_param.o get_var.o ghtcalc.o input_v2_data.o input_v3_data.o int_2_hgt.o  
int_2_prs.o keep_it.o open_file.o os.o pressure_to_height.o proc_v2_data.o  
proc_v3_data.o prscalc.o put_plt_file.o put_var.o pvocalc.o rhucalc.o set_proj.o  
set_proj_v3.o set_vis5d_param.o sfpcalc.o slpcalc.o slpcalc1.o sort_time.o supfields.o  
tdpcalc.o thecalc.o tmr.o trans2vis5d.o tsa.o vervcalc.o virtual.o vis5d_date_time.o  
viscalc.o w.o wdircalc.o wetbulbcalc.o write_data.o wspcalc.o xtdot.o tovis5d.o binio.o  
v5d.o -Vaxlib)  
make[1]: Saliendo directorio `home/mm5v3/TOVIS5D/src'  
rm -f tovis5d ; ln -s src/tovis5d .
```

donde home es la ruta del usuario que está usando el MM5.

Vuelva a editar el fichero **tovis5d.csh** siguiendo las instrucciones del tutorial on-line del MM5.

Ejecute el paquete tecleando:

```
./tovis5d mm5-output-file-name
```

Esta instrucción genera el archivo **tovis5d.log** donde se escriben todos los pasos de la ejecución del módulo TOVIS5D. Edítelo y si el proceso de ejecución fue el correcto, debería aparecer al final del fichero lo siguiente:

```
=====  
=== normally ended ===  
=====
```

### 3.10.2 Conversión al SURFER

Para la utilización de este visualizador, hay que convertir la salida del modelo MM5 a un formato leíble por este programa. Para ello se puede utilizar la opción del programa Vis5D+ llamada, funciones externas o “userfuncs”.

Una vez instalado el Vis5D+ siguiendo el proceso indicado anteriormente, y una vez cargada y visualizada la imagen se procede de la siguiente forma:

Descargarse los siguientes ficheros:

**2SURFp.f**  
**2SURFp.txt**

de la dirección de Internet:

- <http://redibericamm5.uib.es/> (algoritmo n°6)



y copiar estos archivos en el directorio **userfuncs** del Vis5D+ que se ha instalado en la ruta:

**/usr/local/src/vis5d+-1.1**

Edite el fichero 2SURFp.f y cambie lo siguiente (en negrita):

```
#cd /usr/local/src/vis5d+-1.1
#emacs 2SURFp.f &
```

...

```
C look for variables
  do iln=1,maxnl
    p(iln)=1012.5*exp(vert_args(iln)/-7.2)
  enddo
  open(17,file='/usr/local/src/vis5d+-1.1/userfuncs/2SURFp.txt')
C Llegim numero de variables que volem passar a surfer
  read(17,*) numnames
```

...

En lugar de:

...

```
C look for variables
  do iln=1,maxnl
    p(iln)=1012.5*exp(vert_args(iln)/-7.2)
  enddo
  open(17,file='2SURFp.txt')
C Llegim numero de variables que volem passar a surfer
  read(17,*) numnames
```

...

El fichero **2SURFp.txt**, es donde se le indica a esta función externa qué campos y niveles quieren pasarse a SURFER. La estructura del fichero debe ser como la siguiente:

Ejemplo de paso a SURFER de tres campos (este número se indica en la primera línea del fichero: 3 *variables*):

- Geopotencial (H) filtrado dos veces (2 *nfilt*) para dos niveles (2 *levels*) de 1000 y 925 hPa
- Temperatura (T) filtrado dos veces (2 *nfilt*) para 1 nivel (1 *levels*) de 500 hPa
- Humedad Relativa (RH) sin filtrar (0 *nfilt*) para dos niveles (2 *levels*) de 925 y 500 hPa



3 variables

```
=====
"H"
2   nfil
2   levels
1000
925
=====
"T"
2   nfil
1   levels
500
=====
"RH"
0   nfil
2   levels
925
500
=====
```

Para compilar esta función externa

```
#cd /usr/local/src/vis5d+-1.1/userfuncs
#./externf 2SURFp
```

Hay que crear un link desde donde vaya a abrir el fichero **.v5d** hacia el directorio userfuncs del Vis5D+ y crear un directorio que se llame **surferfiles**. Por ejemplo:

```
#cd /home/mm5i/mm5v3/dominios/prueba
#mkdir surferfiles
# ln -s /usr/local/src/vis5d+-1.1/userfuncs userfuncs
```

Después abra el fichero vis5d del dominio de interés, cliquee sobre NEW VARIABLE y del menú que se desplegará cliquee sobre 2SURFp.

En la carpeta **surferfiles** se guardarán los archivos para cada campo y nivel que aparece en el fichero **2SURFp.txt** y para cada hora definida en la salida de simulación hecha por el modelo.

### 3.10.3 Conversión al GrADS (MM5toGrADS)

Descompacte el fichero:

**MM5toGrADS.TAR.gz**

en la carpeta:

**/home/mm5i/mm5v3/MM5toGrADS**



# Manual de instalación del MM5 usando los compiladores FORTRAN de INTEL (IFC)

Diríjase a la ruta anterior y edite el fichero **Makefile**, con los cambios (en negrita) en las siguientes partes del archivo:

```
#cd /home/mm5i/mm5v3/MM5toGrADS
#emacs Makefile &
```

IFC:

```
RM_LIST2 = *.o *.f core .tmpfile make.grads.out grads.namelist grads.print.out
INTEL_LIB = /opt/intel/compiler70/ia32/lib
...
```

```
grep Linux .tmpfile ;\
if [ $$? = 0 ]; then echo "Compiling for Linux" ;\
( $(CD) src ; $(MAKE) all \
"RM = $(RM)" "RM_LIST = $(RM_LIST)" \
"LN = $(LN)" "MACH = linux" \
"MAKE = $(MAKE)" "CPP = /lib/cpp" \
"CPPFLAGS = -I. -C -traditional -DRECLENBYTE" \
"FC = ifc" "FCFLAGS = -FR -pc32 " \
"LDOPTIONS = -i_dynamic" "CFLAGS = -I." \
"LOCAL_LIBRARIES = -L$(INTEL_LIB) -IPEPCF90" ) ;\
```

En lugar de:

PGF:

```
RM_LIST2 = *.o *.f core .tmpfile make.grads.out grads.namelist grads.print.out
```

# Targets for supported architectures

```
...
grep Linux .tmpfile ;\
if [ $$? = 0 ]; then echo "Compiling for Linux" ;\
( $(CD) src ; $(MAKE) all \
"RM = $(RM)" "RM_LIST = $(RM_LIST)" \
"LN = $(LN)" "MACH = linux" \
"MAKE = $(MAKE)" "CPP = /lib/cpp" \
"CPPFLAGS = -I. -C -traditional -DRECLENBYTE" \
"FC = pgf90" "FCFLAGS = -I. -Mfree -O2 -tp p6 -pc \
32 -byteswapio" \ \
"LDOPTIONS = " "CFLAGS = -I." \
"LOCAL_LIBRARIES = " ) ;\
```

Teclee lo siguiente para montar el módulo:

```
make >& make.out &
```

Edite el fichero **make.out** para comprobar que el proceso de montaje ha sido el correcto. Si todo fue bien al final del fichero debería aparecer un mensaje como:

14 Lines Compiled



## Manual de instalación del MM5 usando los compiladores FORTRAN de INTEL (IFC)

---

```
/bin/rm -f writeout.f
ifc -o grads.exe -i_dynamic calclfr.o calcdbz.o calcpt.o calcrh.o calctd.o calcthe.o
calcvordiv.o grads.o interp.o lninterp.o nhgeosig.o tdlinterp.o vect.o seaprsnh.o
dircomp.o pvp.o integrat.o pvs.o advec.o fillit.o writeout.o -
L/opt/intel/compiler70/ia32/lib -IPEPCF90
make[1]: Saliendo directorio `home/mm5v3/MM5toGrADS/src'
```

donde home es la ruta del usuario que está usando el MM5.

Edite el fichero **mm5\_to\_grads.csh**, teniendo en cuenta las instrucciones del tutorial on-line del MM5.

Para ejecutar el módulo, teclee:

```
#!/mm5_to_grads.csh >& log &
```

Edite el fichero **log**. Si todo fue correcto al final del fichero debería aparecer un mensaje tal como:

```
End of file has been detected
```

```
Writing ctl file!
```

```
38 2d variables and 21 3d variables have been written to file
```

```
....Program finished....
```