# WRF\_ARW\_V3.1手册

# 简介

Advanced Research WRF (ARW)模式系统在过去的数年中得到了发展。最近公布了第三版，从2008年4月开始可供使用。ARW是灵活的，最先进的大气模拟系统，它易移植，并且有效的应用于各种操作系统。ARW适用于从米到成千上万公里尺度的各种天气系统的模拟，它的功能包括：

* 理想化模拟（如，LES，对流，斜压波）
* 参数化研究
* 数据同化研究
* 预报研究
* 实时数值天气预报
* 耦合模式应用
* 教学

NCAR的中尺度以及微尺度气象部门最近维护以及支持整个WRF（第三版）的代码的子集，其中包括：

WRF软件框架

Advanced Research WRF (ARW)动力求解方法，包括单向，双向嵌套以及移动嵌套

预处理系统

WRF多种数据同化系统（WRF-Var），该系统还支持3维同化能力

为WRF合作伙伴以及研究论坛提供的数值物理包。一些画图程序和转换适合其他画图工具的程序。这也是本文档的主题。WRF模式系统软件现在是公开以及免费使用的。

如图所示，WRF模式系统包括如下的主要部分：

WRF预处理系统

WRF参数同化系统

ARW求解程序

后处理以及可视化工具

这个程序的主要用于实时数值模拟。其中包括：1）定义模拟区域；2）插值地形数据（如地势，土地类型，以及土壤类型）到模拟区域；3）从其他模式结果中细致网格以及插值气象数据到此模拟区域。它的主要特点包括：

* GRIB 1/2 格式全球格点气象数据
* 地图投影有1）极地投影，2）兰伯特-保角投影，3）麦卡托投影以及4）经纬度投影嵌套
* 由用户接口输入其他数据及met 数据

该程序是可选择的，但可用于将观测数据融入到WPS所产生的插值分析中。它还可以在WRF模式处在循环模式运行时，用于更新WRF模式的初始条件.它的主要特点如下：

它是基于增量变分数据同化技术上而产生出来的

在分析控制变量空间,用共轭梯度法最小化成本函数

非交错的 Arakawa A网格分析

增量分析插值到Arakawa C网格,以及附加到背景场（第一猜测场）以得到WRF模式格点的最终分析

通过使用“obsproc ”工具,可以输入ASCII或者“PREPEUFR”格式的常规观测数据

复合雷达数据（反射率与雷达速度）以ASCII格式输入

背景场（初估计场）的水平分量误差通过递归滤波器（区域范围）或能量谱（全球范围）表现。垂直分量则通过气候学的平均特征向量以及其对应的特征值反映。

水平与垂直背景场误差是不可分割的。每个特征向量的都有各自的水平气候学尺度。

背景场的预处理函数是通过控制变量转换U，其定义为B=UU^T

它包括通过NMC法或整体摄动法，用“gen\_be”程序产生气候学背景场误差协方差估计

在WRF-Var后，用程序更新WRF的边界条件文件

这是模式系统的关键组成部分，它由几个理想化，实时同化以及数值积分的初始化程序组成。它还包括了一个单向嵌套的程序。WRF 模式的关键特征有：

完全可压缩非静力方程带有一个静力选项

区域以及全球的应用

完整的科氏力以及曲率的条件

带有多重嵌套和嵌套层次的双向嵌套

单向嵌套

移动嵌套

地形跟随质量坐标

垂直格点大小随高度变化而变化

地图投影方案  
     极地立体投影（保角映射）  
     兰伯特保角投影  
     麦卡托（保角）投影  
     经纬度旋转投影

Arakawa C格点

Runge-Kutta二阶三阶时间积分

标量保守通量预测变量

二阶到六阶平流选项（水平与垂直）

单调传输以及正定平流选项的水汽，标量，TKE

分时小步长的声波以及重力波模式

小步长水平显显式，垂直隐式

辐散阻尼选项和垂直时间

外部模式滤波选项

上边界吸收以及瑞利Rayleigh 阻尼

侧边界条件

理想化个例：周期边界，对称边界以及开放辐射

真实个例：指定的开放区域

完整的物理选项：陆面，行星边界层，大气与表面辐射，微物理与积云对流

使用上层空气与地表数据以及观测数据格点分析

光谱分析

初始化数字滤波器

重力波拖拽

数个理想化例子

**图形与验证工具**  
该部分包括了：RIP4, NCAR 图形命令语言 （NCL），以及为使用其他作图软件包：GrADS以及Vis5D的转换程序。

VAPOR程序，海洋，大气，太阳研究的可视化以及分析平台，是一个三维数据可视化工具，由NCAR的VAPOR工作组开发与维护。

MET程序，由NCAR的DTC工作组开发与维护。

这些程序的详细信息会在以后的章节进一步介绍。

# WPS系统

介绍  
WRF 前处理系统（WPS）是一个由三个程序组成的模块，这三个程序的作用是为真实数据模拟准备输入场。三个程序的各自用途为：geogrid确定模式区域并把 静态地形数据插值到格点；ungrib从GRIB格式的数据中提取气象要素场；metgird则是把提取出的气象要素场水平插值到由geogrid确定的 网格点上。把气象要素场垂直方向插值到WRF eta层则是WRF模块中的real程序的工作。  
　　上图给出了数据在WPS的三个程序之间的转换关系。正如图像所示，WPS里每个程序都会从一个共同的namelist文件里读取参数。这个 namelist文件按各个程序所需参数的不同分成了三个各自的记录部分及一个共享部分，它们分别定义了WPS系统所要用到的各种参数。被三个程序各自用 到的表格文件没有在图中显示出来。尽管这些表格无需用户改动，但是这些个表格却提供了控制程序运行的额外信息。GEOGRID.TBL, METGRID.TBL,和Vtable文件将会在后文中被详细介绍。

安装WPS的步骤和安装WRF的步骤基本相同，都提供了编译的选项，只是平台有所变化。当MPICH库及合适的库可以使用时，metgird和 geogrid程序可以用分布式内存来编译，如果是这样操作，那当用户在设置大的模拟区域时就可以花更少的时间。但是ungrib程序却不能使用并行，因 此只能用单CPU来操作。  
各个程序的功能; w\* C/ E. N' y" U  
WPS是由三个单独的程序—geogrid，ungrib和metgird组成。当然，也包括了很多其它的应用程序，这些程序放在util目录下。下面是对这三个主要程序的一个简单描述，更详细的内容将在后边的章节进一步介绍。  
程序  
geogrid的目的是确定模拟区域，及把各种地形数据集插值到模式格点上。模拟区域的确定是通过设置namelist.wps文件中的与 “geogrid”有关的参数来实现的。除了计算经纬度和地图每个格点的比例因子外，geogrid还会根据默认值来插值土壤类型、地表利用类型、地形高 度、年平均深层土壤温度、月季植被覆盖、月季反照率、最大的积雪反照率及斜坡的类别。可以通过WRF的官方网站来下载这些场的全球数据集，而且这些[资料](http://www.06climate.com/)可 以被看成是不随时间改变的，因此只需下一次就可以了。很多数据集只有在某一特定精度上才能使用，但是其它的则可在30",2', 5', 和10'中的任何一个精度使用；其中"的意思是弧秒而'则是代表弧分（具体见下面的表格）。尽管一个与模拟区域分辨率接近的地形数据被应用后，这个插值后 的地形场可能更具代表性，但是用户仍是无需下载所有可用的分辨率的数据（SJ认为还是全下载的好）。尽管如此，那些希望应用拥有可以覆盖大范围的格点空间 的区域的用户，还是愿意下载所有分辨率的静态地形数据。  
  
　　除了插值默认的静态数据，geogrid程序还可以插值进更多的连续的且不同种类的地形到模拟区域。可以通过应用表格文件—GEOGRID.TBL来实现插 值新的或额外的数据集到模拟区域。GEOGRID.TBL文件定义了所有可以被geogrid生成的地形场；它描述了插值一个地形场所需的方法，及所需数 据所放的具体位置。由geogrid生成的文件的格式是WRF I/O API，因此可以通过选择NetCDF I/O格式使geogrid生成NetCDF格式的输出文件，以便更方便的用一些外部软件—ncview，NCL和最新版本的RIP4来实现可视化（画出 地形图）。  
  
ungrib程序  
　　ungrib 程序读GRIB文件，“抽出”数据，然后把它们用一个简单的格式写出来，这种格式就是“过渡”格式（格式是SJ的个人翻译，具体参考英文文献，及 writing data to the intermediate format一节以获得关于这个格式的更多细节）。GRIB文件包含随时间变化的气象要素场，而且它是从其它区域或全球模式—如NCEP的NAM或GFS 模式而来的。ungrib程序可以读GRIB1的数据，当然，如果编译时选择了GRIB2的选项，则就可以读GRIB2的数据。  
　　GRIB 文件包含的要素场多于启动WRF所需的数量。两种不同格式的GRIB用了不同的编码来确定变量和在GRIB文件中的层次。ungrib用这些编码表格— Vtable（variable tables）—来确定那些场需要从GRIB文件里提取出来并写成过渡格式。关于编码的细节可以在WMO GRIB文档中找到。对于相同GRIB模式输出文件的各种Vtables可以在WPS主目录下的/ungrib/Variable\_Tables/里。 WPS为NAM104和212格点、NAM AWIP格式、GFS、NCEP/NCAR再分析、RUC（气压坐标数据和混合坐标数据），AFWA的AGRMET地表模式输出，ECMWF及其它数据 集。用户可以以其它Vtable做模板来为其它的模式输出创造自己的Vtable；要进一步了解关于Vtable中要素场的细节，可以参考 creating and editing Vtables。  
　　Ungrib 可以用三个用户可选格式中任何一个来写过渡数据，这三个格式是：WPS—一个新的格式，它包含了对接下来的程序有用的额外信息；SI—WRF之前使用过的 过渡格式；MM5—用来向MM5模式输入GRIB2数据。尽管WPS格式是被推荐使用的，但是这三个中的任何一个格式都可以被用来启动WRF。  
  
程序气  
metgrid程序的作用是把ungrib程序提取出的气象要素场水平插值到geogrid确定的模拟区域上。这个插值后的数据可以被WRF的real程 序所识别并吸收。metgird插值的那些数据的时间段可以通过设置namelist.wps中share记录部分来调整，而且每个模拟区域（最外围区和 嵌套区）的时间都要单独设置。与ungrib程序一样，metgird所处理的数据也是随时间改变的，因此每次做新的模拟时，都要运行metgird程 序。  
　　METGRID.TBL 文件是用来控制如何把气象要素场进行插值的。METGRID.TBL文件为每个要素场都提供了一个区间，在这个区间里，可能会确定诸如要素场的插值方式、 作为标记插值以及要素场所要插值的网格（如ARW的U，V；NMM的H，V）。由metgrid生成的文件的格式是WRF I/O API，因此可以通过选择NetCDF I/O格式使metgrid生成NetCDF格式的输出文件，以便更方便的用一些外部软件—ncview，NCL和最新版本的RIP4来实现可视化（画出 地形图）。  
  
运行WPSMe  
运行WRF Preprocessing System（WPS）有如下三个步骤：  
1，利用geogrid模块确定一个模式的粗糙区域（最外围的范围），及其它嵌套区域  
2，利用ungrib把模拟期间所需的气象要素场从GRIB资料集中提取出来  
3，利用metgird把上述的气象要素场（第二步所做的工作）水平插值到模式区域（第一步所做的工作）中  
当多个模拟在同一区域重复进行时，只需要做一次第一步的工作即可（也就是说geogrid.exe所做出的地形资料—geo\_em.d0\*.nc可以重复 利用）；因此，只有随时间改变的数据才需要在每次模拟时用第二、三步来处理。类似的，如果在多次模拟中，气象数据是类似的，但是地形区域却不断改变的话， 那第二步是可以省略的。下面是各个步骤的详细说明：  
Step 1: 确定模式区域  
如 果WPS安装成功，会在WPS根目录下出现三个可执行程序—geogrid.exe, ungrib.exe和metgird.exe（原程序在各自的同名子目录下）的链接。除了这三个可执行程序的链接，还有一个namelist.wps文 件。下面是一个WPS主目录下各组件的列表：  
  
drwxr-xr-x 2 4096 arch  
-rwxr-xr-x 1 1672 clean气  
-rwxr-xr-x 1 3510 compile  
-rw-r--r-- 1 85973 compile.output  
-rwxr-xr-x 1 4257 configure  
-rw-r--r-- 1 2486 configure.wps  
drwxr-xr-x 4 4096 geogrid  
lrwxrwxrwx 1 23 geogrid.exe -> geogrid/src/geogrid.exe  
drwxr-xr-x 3 4096 metgrid  
lrwxrwxrwx 1 23 metgrid.exe -> metgrid/src/metgrid.exe  
-rw-r--r-- 1 1101 namelist.wps  
-rw-r--r-- 1 1987 namelist.wps.all\_options  
-rw-r--r-- 1 652 namelist.wps.nmm  
-rw-r--r-- 1 4786 README  
lrwxrwxrwx 1 21 ungrib.exe -> ungrib/src/ungrib.exe  
drwxr-xr-x 3 4096 utilMete  
模式的最外层区域和其它嵌套区域都是在namelist.list中geogrid记录里设置的，另外，在share记录里也有需要设置的参数。下面是关 于这两个部分的一个模板，并且如果想获得更多关于每个参数的可能设置及其目的的信息，可以参考description of namelist variables  
&share  
wrf\_core = 'ARW',  
max\_dom = 2,  
start\_date = '2008-03-24\_12:00:00','2008-03-24\_12:00:00',  
; r$ g% N' i\* a  
interval\_seconds = 21600,  
io\_form\_geogrid = 2  
/  
&geogrid  
parent\_id = 1, 1,"   
parent\_grid\_ratio = 1, 3,  
Mj\_parent\_start = 1, 17,  
s\_we = 1, 1,  
s\_sn = 1, 1,  
e\_sn = 61, 97,M  
geog\_data\_res = '10m','2m',  
dy = 30000,!  
map\_proj = 'lambert',  
ref\_lat = 34.83,气  
ref\_lon = -81.03,  
truelat1 = 30.0,  
truelat2 = 60.0,  
geog\_data\_path = '/mmm/users/wrfhelp/WPS\_GEOG/  
/  
为了总结一些列与 geogrid有关的share记录中的典型改变，与WRF的动力核有关的wrf\_core要首先被选择。如果WPS要为ARW（Advanced Research WRF）模拟而运行，那wrf\_core就设成‘ARW’，如果要为NMM（Nonhydrostatic Mesoscale Model）模拟的话，则设成‘NMM’。当选择好动力内核后，接下来选择max\_dom，即区域（最外层的一个 嵌套数）的总数（当 wrf\_core=‘ARW’）或者嵌套的层次（当wrf\_core=‘NMM’）。因为geogrid生成的仅仅是时间独立的数据，因此 start\_date,end\_date, 和 interval\_seconds这些参数将被其忽略。另外，还有一些可选的选项，如opt\_output\_from\_geogrid\_path，如果设 成默认值，则由geogrid.exe生成的地形文件将被放到当前工作目录（WPS的主目录），如果想放到别的目录下，则根据需要修改即 可；io\_form\_geogrid则是设置地形数据输出格式的。在geogrid的记录部分，是关于模拟区域投影的设置，同时也设置了模式格点的大小和 所在位置。模式所用的地图投影方式由map\_proj来设置，其它用来设置投影的参数总结如下：Meteorological Numerical Model Union of China (MNMUC) 中国气象数值模式联盟. \_/ g# |& A& Y0 M  
地图投影方式/与相应投影方式有关的参数变量  
'lambert' truelat1  
truelat2 (optional)  
stand\_lon& C6  
'mercator' truelat1  
'polar' truelat1  
stand\_lon  
'lat-lon' pole\_lat  
stand\_lon  
如果WRF是在一个局地区域里运行，那粗糙区域（最外层区域）位置则是通过ref\_lat和ref\_lon来定位，它们分别确定了粗糙区域的纬度和经度。如 果也要处理嵌套区域，则它们的位置是通过i\_parent\_start 和j\_parent\_start来确定；更多关于设置嵌套区域的细节可以参考nested domains一节。接下来，粗糙区域的维数由dx和dy来确定，它们分别确定了x轴和y轴上标准格距的长度，而e\_we和e\_sn则分别给出了x轴（东 西方向）和y轴（南北方向）上的格点数（具体设置方案可见后边description of namelist variables）；对应'lambert', 'mercator', 和 'polar'投影方式，dx和dy的单位是'米'，对于'lat-lon'投影方式，dx和dy的单位则是'度'。对于嵌套区域，只有e\_we和 e\_sn可以用来确定格点的维数，格点的dx和dy是不能被设定的，因为它们的值已经被parent\_grid\_ratio和parent\_id所决定 了，这两个参数分别确定了嵌套的上一级区域（父区域）的格点距离与嵌套的格点距离的比值和嵌套的上一级区域的ID【】。  
对于全球的模拟，粗糙区域（最外围区域）的覆盖范围就应该是全球，所以ref\_lat和ref\_lon就不在被使用，而且dx和dy也不应该再被设置，这 是因为这个格距将根据格点数被自动计算出来。同样需要注意的是，经纬度投影（map\_proj = 'lat-lon'）是WRF中唯一支持全球区域（全球模拟）的投影方式。  
除 了设置与模式区域的投影方式、位置和覆盖范围有关的参数外，静态数据集的路径也必须通过参数geog\_data\_path被正确的设置。同样，用户可能会 通过设置geog\_data\_res参数来选择静态数据的分辨率以便于geogrid的差值，而分辨率的值要与GEOGRID.TBL中的数据分辨率吻 合。如果静态数据集是从WRF官网下载的，则可能的分辨率会包括'30s', '2m','5m', 和 '10m',分别对应的是30秒，2-，5-，10-分的数据【】。根据wrf\_core所设值的不同，geogrid需要不同的GEOGRID.TBL 与之对应，因为WPS会根据动力内核的不同插值出不同的网格。如果wrf\_core='ARW'，则应该使用GEOGRID.TBL.ARW，如果是 wrf\_core='NMM'，则相应的要用GEOGRID.TBL.NMM。可以通过链接正确的GEOGRID.TBL文件来正确使用它，原始的文件放 在geogrid目录里（另一种方法是通过opt\_geogrid\_tbl\_path参数来设定文件的位置来使用它）。  
$ ls geogrid/GEOGRID.TBL  
lrwxrwxrwx 1 15 GEOGRID.TBL -> GEOGRID.TBL.ARW  
如果想了解更多的关于各个参数的意义和可能的选项，可以参考description of the namelist variables。  
在namelist.wps中设置了合适的模拟和嵌套区域，geogrid.exe就可以处理地形文件了。在ARW的个例中，地形文件被命名为 geo\_em.d0N.nc，其中每个文件中的N代表嵌套的序号。当运行NMM个例时，粗糙区域（外围区）的地形数据文件的名字为 geo\_nmm.d01.nc，嵌套的文件则是geo\_nmm\_nest.l0N.nc，其中N代表嵌套的级数。还需要注意的是，文件的后缀会随参数 io\_form\_geogrid设定的不同而不同。运行geogrid.exe的命令是：  
$ ./geogrid.exe% J# g$ h/ f4 o% u9 I) \_  
当程序运行成功后，会出现如下信息气  
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!" u/ {- {6 w" m6 H  A1 V6 W, H E  
! Successful completion of geogrid. !  
并 且WPS的主目录下会出现地形文件（或者出现在opt\_output\_from\_geogrid\_path所设定的目录下）。如果没有出现，可以通过检查 geogrid.log来找出可能的失败原因。如果想了解更多的关于检查geogrid输出结果的信息，可以参考checking WPS output一节。  
$ls  
drwxr-xr-x 2 4096 arch#   
-rwxr-xr-x 1 1672 clean  
-rwxr-xr-x 1 3510 compile  
-rw-r--r-- 1 85973 compile.output  
气象-rw-r--r-- 1 2486 configure.wps  
-rw-r--r-- 1 4745324 geo\_em.d02.nc  
drwxr-xr-x 4 4096 geogrid  
lrwxrwxrwx 1 23 geogrid.exe -> geogrid/src/geogrid.exe  
-rw-r--r-- 1 11169 geogrid.log9、  
-rwxr-xr-x 1 1328 link\_grib.csh   
drwxr-xr-x 3 4096 metgrid  
-rw-r--r-- 1 1094 namelist.wps  
-rw-r--r-- 1 1987 namelist.wps.all\_options   
-rw-r--r-- 1 1075 namelist.wps.global  
-rw-r--r-- 1 652 namelist.wps.nmm  
drwxr-xr-x 4 4096 ungrib  
lrwxrwxrwx 1 21 ungrib.exe -> ungrib/src/ungrib.exe   
drwxr-xr-x 3 4096 util  
Step 2:利用ungrib从GRIB文件中提取气象要素场  
当已经下载了GRIB格式的气象数据后，提取要素场以转成过渡格式的第一步所要做的参数设置涉及到“share”和“ungrib”两个部分。下面是关于这两部分的一个模板：  
&share  
wrf\_core = 'ARW',  
start\_date = '2008-03-24\_12:00:00','2008-03-24\_12:00:00',  
interval\_seconds = 21600,  
io\_form\_geogrid = 2  
/  j  
&ungrib) s) h8 D) V. ~9 ?  
out\_format = 'WPS',  
/  
在 “share”部分，与ungrib有关的参数是粗糙区域（外围区）的开始和结束时间（start\_date和end\_date; 或者start\_year, start\_month, start\_day, start\_hour, end\_year, end\_month, end\_day和end\_hour）以及气象数据文件的时间间隔（interval\_seconds）。在“ungrib”部分，参数 out\_format是用来设置过渡数据文件的格式的；metgird程序可以读任何一个被ungrib支持的格式，因此虽然'WPS'是被推荐的格式, 但是'WPS'，'SI', 和'MM5'中的任何一个都可以作为选项被选择；另外，用户可以通过设置prefix参数来确定过渡数据文件的保存路径和前缀。例如，如果prefix被 设成'ARGRMET'，那过渡数据的文件名就是AGRMET:YYYY-MM-DD\_HH，其中YYYY-MM-DD\_HH是文件中数据的真实的日期。  
在合理的修改了 namelist.wps文件后，Vtable文件也是需要事先被提供的，并且GRIB文件也要被链接（拷贝）到ungrib.exe所期望的目录下（一 般是WPS的主目录，如果WPS被安装成功的话）。如果WPS要处理气象数据，就要在Vtable的辅助下进行才行，即使Vtabel仅仅被象征性地以 Vtable来命名并链接到ungrib.exe所期望的目录下（一般是WPS的主目录）。例如，如果是来自GFS模式的GRIB数据，可以如此处理：  
$ ln -s ungrib/Variable\_Tables/Vtable.GFS Vtable  
程序将尝试以 GRIBFILE.AAA,GRIBFILE.AAB, …, GRIBFILE.ZZZ这样的文件名来读取GRIB数据。为了能简化把GRIB数据链接到到合适的目录，并以这些名字来命名，程序提供了一个shell 的脚本link\_grib.csh。这个脚本通过读取一个以GRIB数据文件名为列表的命令行参数来完成上述工作。例如，如果GRIB数据被下载到一个 /data/gfs的目录下，那这些文件可以用下面的方法来进行链接：  
$ ls /data/gfs/ j. ~! ~\* G% Z  
-rw-r--r-- 1 42728372 gfs\_080324\_12\_00  
-rw-r--r-- 1 48218303 gfs\_080324\_12\_06  
$ ./link\_grib.csh /data/gfs/gfs\*  
上述操作完成后，WPS目录下的列表将是如下所示：  
drwxr-xr-x 2 4096 archM  
-rwxr-xr-x 1 1672 clean  
-rwxr-xr-x 1 3510 compile  
-rwxr-xr-x 1 4257 configure  
-rw-r--r-- 1 2486 configure.wps  
-rw-r--r-- 1 1957004 geo\_em.d01.nc  
drwxr-xr-x 4 4096 geogrid  
lrwxrwxrwx 1 23 geogrid.exe -> geogrid/src/geogrid.exe  
-rw-r--r-- 1 11169 geogrid.log  
lrwxrwxrwx 1 38 GRIBFILE.AAB -> /data/gfs/gfs\_080324\_12\_06  
-rwxr-xr-x 1 1328 link\_grib.csh" s D9 H3 v: d4 K  
drwxr-xr-x 3 4096 metgrid  
-rw-r--r-- 1 1094 namelist.wps  
-rw-r--r-- 1 1987 namelist.wps.all\_options  
-rw-r--r-- 1 1075 namelist.wps.global  
-rw-r--r-- 1 652 namelist.wps.nmm  
drwxr-xr-x 4 4096 ungrib  
lrwxrwxrwx 1 21 ungrib.exe -> ungrib/src/ungrib.exe  
drwxr-xr-x 3 4096 util  
lrwxrwxrwx 1 33 Vtable -> ungrib/Variable\_Tables/Vtable.GFS  
修改完namelist.wps并链接了合适的Vtable和正确的GRIB文件后，ungrib.exe就可以被执行以生成过渡数据格式的气象数据文件了。Ungrib可按如下命令进行操作：气象,  
数值$ ./ungrib.exe >& ungrib.outputMet  
因为ungrib程序会产生一个数量相当可观的输出，因此这个操作是被推荐的，这样可以把输出间接地输入到一个名字为ungrib.output的文件中。如果ungrib.exe运行成功，过渡数据文件将会在正确的目录中出现，且则会有如下信息被打印在文件的最后部分：  
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!: K/ ? P% u# |# i/ b' z  
! Successful completion of ungrib. !  
B, n# Z! O- B\* {# K) ^% R这个过渡数据的文件名将会以FILE:YYYY-MM-DD\_HH的形式命名（除非prefix参数被设置成其它名字）  
$ls  
drwxr-xr-x 2 4096 archM  
-rwxr-xr-x 1 1672 clean  
-rwxr-xr-x 1 3510 compile  
-rw-r--r-- 1 85973 compile.output  
-rw-r--r-- 1 2486 configure.wps  
-rw-r--r-- 1 154946888 FILE:2008-03-24\_12  
-rw-r--r-- 1 154946888 FILE:2008-03-24\_18气象  
-rw-r--r-- 1 1957004 geo\_em.d01.nc  
-rw-r--r-- 1 4745324 geo\_em.d02.nc7 v&  
drwxr-xr-x 4 4096 geogrid  
lrwxrwxrwx 1 23 geogrid.exe -> geogrid/src/geogrid.exe  
-rw-r--r-- 1 11169 geogrid.log  
lrwxrwxrwx 1 38 GRIBFILE.AAA -> /data/gfs/gfs\_080324\_12\_00  
lrwxrwxrwx 1 38 GRIBFILE.AAB -> /data/gfs/gfs\_080324\_12\_06  
drwxr-xr-x 3 4096 metgrid  
lrwxrwxrwx 1 23 metgrid.exe -> metgrid/src/metgrid.exe  
-rw-r--r-- 1 1987 namelist.wps.all\_options  
-rw-r--r-- 1 1075 namelist.wps.global  
-rw-r--r-- 1 652 namelist.wps.nmm  
-rw-r--r-- 1 4786 README  
drwxr-xr-x 4 4096 ungrib  
lrwxrwxrwx 1 21 ungrib.exe -> ungrib/src/ungrib.exeM  
-rw-r--r-- 1 1418 ungrib.log  
,drwxr-xr-x 3 4096 util

气ungrib/Variable\_Tables/Vtable.GFSMete  
WPS创建嵌套区域气象  
做带嵌套的模拟试验并不比做单层区域的难多少，嵌套区域的模拟和单层嵌套的不同之处在于，当运行geogrid和metgrid程序时，处理的将不止一个格点了。为了确定嵌套的大小和位置，将要设置一系列的namelist.wps中的参数，下面是设置一个嵌套的模板：  
&share  
wrf\_core = 'ARW',气象  
max\_dom = 2,Meteo   
start\_date = '2008-03-24\_12:00:00','2008-03-24\_12:00:00', 、  
end\_date = '2008-03-24\_18:00:00','2008-03-24\_12:00:00',  
interval\_seconds = 21600,  
io\_form\_geogrid = 2  
Me  
&geogrid  
parent\_id = 1, 1,  
parent\_grid\_ratio = 1, 3,  
j\_parent\_start = 1, 17,气象  
s\_we = 1, 1,  
s\_sn = 1, 1,  
e\_sn = 61, 97,  
geog\_data\_res = '10m','2m',  
dx = 30000,  
dy = 30000,  
map\_proj = 'lambert',  
ref\_lat = 34.83,M  
ref\_lon = -81.03,  
  
truelat2 = 60.0,  
stand\_lon = -98.  
/  
设置 与嵌套有关的参数如上。在上边这个例子中，“share”中第一个要改的是max\_dom的数值，即总共的区域个数（最外围区域 嵌套区）。由此，以后任 何与区域有关的参数都要设置N个值，每个值对应一个网格。“share”中唯一需要改变的是模拟的起止时间。每个模拟区域都要有一个起止时间与之对应，当 然这个时间的设置是有限制的，那就是子区域的起始时间一定要晚于或至少与其父区域（上一级区域）起始时间相等；同样，终止时间也一定要早于或至少与父区域 的终止时间相等；而且，嵌套的起止时间的给定最好都和用户想要嵌套启动的时间一致（即起始时间是嵌套的起始时间，并且终止时间也等于起始时间）。这是因 为，嵌套是从它的父区域得到的边界条件，因此，除了WRF中用到网格松弛，或者说松弛分析，嵌套只要在初始时间接受WPS的处理即可。但是很重要的一点 是，当运行WRF时，对每个区域的真实的起始时间却是要在namelist.input文件里详细给出的。  
剩下要做的就是对geogrid 进行设置了。在这块记录中，通过改变参数parent\_id来设置每个区域的父区域，除了粗糙区域（最外围区）的父区域是自己本身外，每个区域都必须是另 一个区域的子区域。与每个区域的父区域的ID类似，每个父区域和其子区域的格点距离的比值是通过参数parent\_grid\_ratio来设定的。

接下来，需要通过参数i\_parent\_start和j\_parent\_start来设定子区域左下角在父区域内的位置（i，j），并且这个位置是在无参差格点的情况下确定的。每个区域的格点维数都要通过参数s\_we, e\_we,s\_sn,和e\_sn来设置。关于与格点有关的设置的模板在上图已经给出。其中，每个网格在南北（s\_sn）向和东西（s\_we）向的起始格点 都要设置成1，而相应的终止（e\_sn和e\_we）格点则是区域的总维数，为了保证每个嵌套区域的右上角与父区域的无参差格点想重合， e\_sn和e\_we的大小必须要比parent\_grid\_ratio所设数值的N倍再大一个。同样的，每个区域所要插值的源数据的分辨率（可参考 description of namelist variables）需要通过参数geog\_data\_res来设定。用户可通过阅读description of namelist variables获得一个对namelist各个参数的完整描述。

metgrid程序可以插值不随时间变化的要素场，也能从多个源数据中获得要素场并进行插值。第一个功能可以通过设置&metgrid记录中的参 数constants\_name来实现，这个参数可以用来设置一系列具有过渡格式的文件的文件名（如果需要还可包含路径信息），这些文件里面包含了不随时 间改变的要素场，并且在每次metgrid进行处理时都可以被用上，例如，短时模拟可能会用上常值SST（海表温度）场—这个场只需有一个时次的值就可以了，用法如下：  
&metgrid  
/   
或者  
; o8 Cconstants\_name = 'LANDSEA', 'SOILHGT  
/  
第二个功能—从多个源数据里获得资源进行插值—在有两个或两个以上补充数据（随时间变化）且需要被合并到一起后进行real.exe处理的情况下是非常有 用的。可以通过&metgrid记录中的参数fg\_name设置一系列具有过渡格式的数据的前缀（如果需要还可包含路径信息）来实现此功能。当给 出多个路径-前缀的设置，如果相同的气象要素场在多个数据文件中出现，则最后一次出现的数值将被使用，不论之前出现过多少次—靠后原则。因此，可以通过调 整源数据的顺序来安排它们的优先级别。下面是一个关于使用插值地面要素场和高空风场到地形网格上的模板：  
&metgrid  
fg\_name = '/data/ungribbed/SFC', '/data/ungribbed/UPPER\_AIR'  
可 以通过合理的设置&ungrib记录中的参数prefix来达到简化从GRIB文件中提取要素场的步骤。这个参数允许用户控制ungrib程序所 生成的过渡文件的路径和文件名。下面将给出这种方法在应用上的一个模板。North American Regional Reanalysis (NARR)—北美区域再分析资料集被分为了三部分，分别是三维的大气数据，地面数据和常值场数据，如果用户想使用这种数据，可以通过脚本 link\_grib.csh把3-D GRIB数据进行链接，然后再链接NARR Vtable到文件Vtable（具体方法见ungrib程序一节），然后在运行ungrib之前通过设置&ungrib记录里的prefix使 得产生的过渡文件有一个合适的前缀，模板如下：  
&ungrib  
Met  
prefix = 'NARR\_3D',  
/  
运行完ungrib.exe后，将会出现如下的文件（附带正确的日期）:  
NARR\_3D:2008-08-16\_15M  
NARR\_3D:2008-08-16\_18  
  
考虑到过渡文件是一个三维场的数据，因此当处理地面数据时可以另外改一个prefix与之对应:  
out\_format = 'WPS',/ [  
prefix = 'NARR\_SFC',  
/  
再一次运行ungrib.exe，会另外出现如下的数据文件:气象,  
NARR\_SFC:2008-08-16\_12  
NARR\_SFC:2008-08-16\_15  
NARR\_SFC:2008-08-16\_18  
...  
最后，当处理常值数据时，同样要给出合适的prefix:  
out\_format = 'WPS',  
prefix = 'NARR\_FIXED',  
/  
第三次运行ungrib.exe后，会得到如下数据文件:  
NARR\_FIXED:1979-11-08\_007 ^2 @9 A5 Q2 `# y  
为了清楚起见，常值数据最好省掉日期信息。如，因为数据是静态的，所以可以把上面提到的文件名改为NARR\_FIXED。在这个例子中，NARR常值数据 仅仅在一个特定的时次—1979年11月08日00时00分（UTC）可以使用，因此，当运行ungrib程序之前，用户应该设置好&share 记录中的起止日期；当然，在运行metgrid程序之前，这个日期还需再次被调整。  
考虑到NARR数据集有三个过渡文件，因此在运行metgrid.exe之前需要设置好&metgrid记录里的相关参数—constants\_name和fg\_name，模板如下：  
&metgrid3 D3 v# Y9 h\* N  ^. @4 f6 E0 @  
constants\_name = 'NARR\_FIXED',  
4/  
尽管 可能性很小，但是还是会发生的一种情况是，当一个来自区域模式的气象源数据不足以覆盖整个模拟区域时，则多个源数据将被使用，并且那些来着一个更大的区域 模式或全球模式的数据将被插值到剩下的模拟网格上。例如，为了能在有可能的区域上应用NAM的数据，剩下的区域应用GFS的数据，需要设置 namelist中的如下参数：  
&metgrid  
fg\_name = '/data/ungribbed/GFS', '/data/ungribbed/NAM'  
/  
则插值后的结果如下图：  
  
如果要素场不会在其它源数据文件中出现，那就不会有优先级一说，则metgrid进行的插值就会像正常情况下一样，每个源数据都要覆盖整个模拟区域以免有的区域没有数据。  
WPS中并行  
如 果WPS中设置的模拟区域的维数过于庞大以至于单CPU无法胜任，则用分布式存储器的方式来运行geogrid和metgrid程序是一个解决办法。为了 能把geogrid和metgrid程序以分布式存储器的类型编译，用户必须在机器上安装MPI库，及在编译时选中选项“DM parallel”。当编译成功后，geogrid和metgrid程序就可以在mpirun和mpiexec的命令，或通过批队列系统下运行了，具体的 运行方式则是根据机器的不同而定。正如之前提到的，ungrib程序是不能用并行运行的，而且ungrib程序对于内存的需求是独立于geogrid和 metgrid程序之外的；因此，无论编译时是否选择“DM parallel”选项，ungrib总是会在单处理器下编译且在单CPU下运行。  
每 个标准的WRF I/O API 格式 (NetCDF, GRIB1, binary)都有一个相应的并行格式，但这需要在设置参数io\_form（即io\_form\_geogrid和io\_form\_metgrid）时在原 有的值的基础上加100来实现。当并行被使用时，每个CPU都会对其读/写的输入/输出的文件进行分割，这些文件的名字与在串行时的一样，只是多了一个4 位数的处理器ID来代表处理器的名字。例如，当用4个处理器来运行geogrid（io\_form\_geogrid=102）时，单粗糙区域（最外围区） 就可以生成四个输出文件，分别是geo\_em.d01.nc.0000, geo\_em.d01.nc.0001, geo\_em.d01.nc.0002和geo\_em.d01.nc.0003。在并行的过程中，模式区域会被分成若干矩形区域，每个处理器会分派一个单 独的子区域，当读/写到WRF I/O API格式时，每个处理器也都只会读/写它自己的那块子区域。因此，如果geogrid程序使用并行，则metgrid运行时所使用的处理器个数要与 geogrid时的一样。同样的，如果metgrid使用并行运行时，则real.exe也需要用相同的处理器个数来运行。当然，在用多个处理器运行时，还是可以用一个标准的输入输出格式的，此时，模式区域的所有数据都会被分配/收集到输入/输出中。最后，当用多个处理器运行geogrid或者 metgrid时，每个处理器都会写一个独立的log文件（类似工作日记的文件），这些log文件的名字会附上4位数的处理器ID，以此作为I/O API文件。  
检查WPS的输出  
当 运行WPS时，检查它们的输出是有帮助的。例如，当决定了嵌套的位置后，检查下被插值的静态地理数据和经纬度场是有益的；另一个例子，当向WPS输入新的 源数据时—无论是静态数据还是气象要素场—检查插值后的结果对通过geogrid和metgrid调整插值的方法也是有好处的。由于使用NetCDF格式 作为geogrid和metgrid的输入输出格式，使得一些列可以读NetCDF数据的可视化工具都可以用来检查那些被geogrid处理和 metgrid水平插值的气象要素场的区域文件。可以通过设定参数io\_form\_geogrid和io\_form\_metgrid为2来让 geogrid和metgrid的结果以NetCDF格式输出（2是这两个参数的默认设置）：  
  
io\_form\_geogrid = 2,  
/  
io\_form\_metgrid = 2,% ?  
/  
在这些可用的工具中，ncdump，ncview和新的RIP4程序是比较好的。Ncdump是一个具有NetCDF库的可以把NetCDF数据的变量及 属性显示出来的简洁程序。在检查geogrid生成的区域文件中的区域参数（如东西、南北向维数，或区域中心点），或展示一个文件中各要素场时都是有帮助 的。而ncview则提供了一个看NetCDF文件中各要素场的互动式的方法。如果用户想生成可以用在刊物上发表的各要素的图片，那新版的RIP4则是有 用的。新版RIP4不仅可以画水平等值线、地图背景，还可以在同一幅图上还可以叠加很多要素场。从ungrib程序中输出的文件是以简单的二进制格式 （‘WPS’, ‘SI’, 或 ‘MM5’）写成的，所以大部分用于看NetCDF文件的软件都法使用了。尽管如此，一个以NCAR Graphics为基础的应用软件—plotfmt（包含在WPS内核中）可以解决这个问题。这个应用程序制作出的要素场的等值线可以通过一个多媒体软件 来查看。如果NCAR Graphics库被正常安装了，则当编译WPS时，plotfmt会和其它一些应用程序一起被自动编译。0 k& Q) x# ]& ^, D  S, o  
WPS应用程序  
处理WPS的三个主要程序—geogrid，ungrib和metgrid—WPS还有很多其它的应用程序会被编译，并放在util目录下。这些程序可以被用来检查数据文件，看模拟区域的位置，计算气压场及计算平均的地表温度场。- x% F, t4 Z' l1 J, Y: k# [  
A． avg\_tsfc.exe  
程序是用来计算日平均地表温度的，这些数据来自于过渡格式文件，而它们的日期则是通过namelist.wps中的“share”记录中的相关参数定的， 并且同样考虑到了文件之间的时间间隔，在计算平均值时，avg\_tsfc.exe以namelist.wps中设定的起始时间为准，并用上尽可能多时次的 数据。如果一个中天没有一个完整的时次，则不会有输出文件生成，这个程序也会立即停止。类似的，如果一个文件的时次没有包含在一个完整的24小时的周期 内，则这个文件也将会被舍掉，例如，如果有5个过渡文件，之间的时间间隔为6个小时，则最后一个文件将被舍弃。计算出的平均场将以文件的形式输出，名字为 TAVGSFC，格式同样是过渡格式。日平均的地表温度可以被metgrid所使用，方法是在“metgrid”记录部分中设置 constants\_name=“TAVGSFC”。  
B． mod\_levs.exe气象  
mod\_levs.exe程序被用来移除过渡格式中数据的层次。这个层次将会保存在新的namelist.wps中的相关部分：  
press\_pa = 201300 , 200100 , 100000 ,  
95000 , 90000 ,  
85000 , 80000 ,  
75000 , 70000 ," c! x- }$ T\* e3 \_  
65000 , 60000 ,  
45000 , 40000 ,  
35000 , 30000 ,  
25000 , 20000 ,! Z  
15000 , 10000 ,:  
5000 , 1000  
气象,  
在&mod\_levs记录部分中的参数press\_pa是用来保存层次列表的，这些确定的层次要与过渡格式文件中的xlvl（具体参考WPS intermediate format）的值相匹配。mod\_levs程序要附带两个命令行参数作为它的输入。第一个参数是要处理的过渡文件的名字，第二个是输出文件的名字。从气 象数据集中移除不需要的层次是很有用的。例如，当一个数据集被用来作为模式的初始条件，另一个则作为边界条件，可以通过在第一个时间周期中用 metgrid的插值来提供初始条件，而边界条件的数据集则用于其它时次来实现。如果两个数据集的垂直层次相同，则无需再做其它工作，但是当它们的层次数 不同，则必须以最小的层次为标准，在有m层数据的文件中移除（m-n）个层次，这里m>n,并且m和n是两个数据集中各自的层次数。必须有相同层次 的要求来自于real.exe的限制，它要求必须垂直层次必须是常值以便从中进行插值。  
的应用只 是处理两个或更多具有不同垂直层次的数据集的一个临时解决方法。如果用户选择使用mod\_levs，则需要注意的是，尽管数据集之间垂直层次的设定可能不 同，但是所有数据都要有个地表的层次，并且当运行real.exe和wrf.exe时，参数p\_top（namelist.input）必须被设定成所有 数据集中最低层次以下的某一层。. o5 c4 |& q) G B) s2

C．        calc\_ecmwf\_p.exe8 n M, c) v2 p/ Y, d& y' A, Y- Q  
在垂直插值气象要素场的过程中，real程序要求在每个相同的层次上和其它气象要素一样都有一个3-D的气压场。calc\_ecmwf\_p.exe应用程序可以被用于利用ECMWF的sigma坐标数据集来生成这样一个气压场。考虑了地表气压场（地面气压场的记录）及一些列系数A和 B，calc\_ecmwf\_p.exe可以在ECMWF的sigma坐标中的k层的（i，j）点计算出气压，即Pijk = Ak Bk\*Psfcij。这些系数可sigma高度-系数的对应表中找到。这个文件被写在普通的txt文件，ecmwf\_coeffs中。例如，16个sigma层的emcwf\_coeffs文件会包含以下内容：  
0 0.000000 0.000000000" k\*   
1 5000.000000 0.000000000M  
2 9890.519531 0.001720764  
3 14166.304688 0.013197623  
4 17346.066406 0.042217135  
5 19121.152344 0.093761623  
! w  
7 18164.472656 0.268015683  
9 12488.050781 0.510830879  
10 8881.824219 0.638268471  
11 5437.539063 0.756384850  
  
13 783.296631 0.928746223  
14 0.000000 0.972985268  
15 0.000000 0.992281914  
16 0.000000 1.000000000  
考虑到ungrib生成的过渡格式文件和ecmwf\_coeffs文件，calc\_ecmwf\_p会在namelist.wps定义的整个时间周期中不停循 环，且每个时次都会生成一个额外的过渡格式文件PRES:YYYY-MM-DD\_HH，文件里包含了完整的sigma层次和3-D的相对湿度场。这个过渡 文件可以通过在参数fg\_name中添加‘PRES’使之与ungrib生成的过渡文件一起被metgrid处理。  
D． plotgrid.exe  
是一个以NCAR Graphics为基础的应用程序，它的作用是画出namelist.wps中定义的模拟区域的位置。程序对namelist.wps进行操作，因此无需运行WPS的三个主程序。成功完成后，plotgrid生成了一个NCAR-G的元文件，gmeta，这个文件可以被idt所查看。粗糙区域（最外围区） 占据整个画面，一个地图轮廓包围了粗糙区域，并且如果有嵌套区域则会以矩形框的形式被标注在最外围区里（见下图）。这个程序在最初设定模拟区域或每次调整 模拟区域位置时很有用。目前，这个程序还不能画出ARW中用lat-lon投影方式（map\_proj = 'lat-lon'）设置的区域。  
E． g1print.exe  
g1print.exe唯一的一个命令行参数是GRIB1的文件名。程序会画出文件中要素场，层次和数据的日期。下面是一个fnl文件的模板：  
Copen: File = fnl\_080801\_00\_00                                                                                                      
Fortran Unit = 0  
  
rec GRIB GRIB  Lvl  Lvl  Lvl         Time      Fcst气  
Num Code name  Code one  two                   hour  
----------------------------------------------------  
   2   7 HGT    100  975    0  2008-08-01\_00:00 00  
   3   7 HGT    100  950    0  2008-08-01\_00:00 00  
   4   7 HGT    100  925    0  2008-08-01\_00:00 00  
% \4 h, ^   
  286 230 5WAVA  100  500    0  2008-08-01\_00:00 00Met  
  
\*\*\*\*\* End-Of-File on C unit   3  
F.g2print.exe. s( ]: A$ Y/ O7 c0 E. v  
与g1print.exe类似，只是它针对的是GRIB2的数据文件。  
G．plotfmt.exe  
plotfmt.exe 是一个NCAR-G的程序，它画的是过渡格式文件的内容。程序唯一的命令行参数是要画的文件名，并且生成NCAR-G的元文件，这个文件里包含了输入文件 中各个要素场的等值线图。graphics的源文件，gmeta，可以用idt来查看，并且用ctrans来转换成其它格式。下面是一个模板：  
H．rd\_intermediate.exe  
rd\_intermediate.exe可以把过渡格式文件中各要素的信息都打印出来。  
================================================  
UNITS = Pa DESCRIPTION = Sea-level Pressure  
DATE = 2008-08-09\_00:00:00 FCST = 0.0000005 ]9  
SOURCE = NCEP GFS GDAS/FNL       GRID   3  
LEVEL = 201300.000000) a" ^3 h' {3 h: J/ i  
I,J DIMS = 360, 181  
IPROJ = 0  
  REF\_X, REF\_Y = 1.000000, 1.000000  
  REF\_LAT, REF\_LON = 90.000008, 0.000000  
  DLAT, DLON = -1.000000, 1.000000气  
DATA(1,1)=101440.000000  
! x& p9 ^- ?) |$ b" O! X  
================================================\* T; v0 x& i5 t  
…………………………………  
[2 N7 I' m" \_: a/ \7 o' J  
FIELD = GHT  
  
DATE = 2008-08-09\_00:00:00 FCST = 0.000000  
SOURCE = NCEP GFS GDAS/FNL       GRID   38  
LEVEL = 1000.000000  
I,J DIMS = 360, 181  
IPROJ = 0  
  REF\_X, REF\_Y = 1.000000, 1.000000  
  
  DLAT, DLON = -1.000000, 1.000000  
DATA(1,1)=31929.201172  
SUCCESSFUL COMPLETION OF PROGRAM RD\_INTERMEDIATE  
  
把气象要素场写成过渡格式  
ungrib的作用是读取GRIB数据集并写成一个简单的过渡格式，这个格式是可以被metgrid所识别的。如果某气象要素并不存在于GRIB1/2格 式中，则用户就有必要把这类数据写入过渡格式文件中。幸运的是，过渡格式是一种相对简单的并且顺序固定的，且是由Fortran写成的无格式数据。需要注 意的是，这些无格式写入是用得big-endian byte order（最高位字节在最前头的排列顺序），这种写入方式可以由编译所识别。下面来描述一下WPS的过渡格式；对SI或MM5的过渡格式熟悉的用户会非 常容易熟悉WPS格式，因为它是非常类似的，而且WPS格式可以检查Fortran的子程序，这些子程序（分别为metgrid/src /read\_met\_module.F90和metgrid/src/write\_met\_module.F90）可以读写所有的三种的过渡格式。当在写 成WPS过渡格式时，2-D的要素场作为一个实型矩形数组来写的。3-D的数组则会以垂直维数为基础分成多个2-D数组，然后被独立写入。需要注意的是， 全球数据集—要么是高斯投影，要么是等距圆柱投影，而局地数据集—要么是麦卡托投影，要么是兰伯托投影，再就是极射投影或等圆柱体投影。写的顺序用于在 WPS过渡格式中写一个单独的2-D数组，正如下面所示（注意：并不是所有定义的变量都是用来做一个给定数据的投影）：) ?# P. q  
integer :: version ! Format version (must =5 for WPS format)Me  
integer :: nx, ny ! x- and y-dimensions of 2-d array  
integer :: iproj ! Code for projection of data in array:  c g9 n! {3 e) ]" S/ K) }/ a  
! 0 = cylindrical equidistant  
! 1 = Mercator  
! 3 = Lambert conformal conic;  
! 4 = Gaussian (global only!)  
! 5 = Polar stereographic x(  
real :: nlats ! Number of latitudes north of equator  
! (for Gaussian grids)  
real :: xfcst ! Forecast hour of data  
real :: xlvl ! Vertical level of data in 2-d array  
real :: startlat, startlon ! Lat/lon of point in array indicated by  
! startloc string  
  
real :: dx, dy ! Grid spacing, km  
real :: xlonc ! Standard longitude of projection6  
real :: truelat1, truelat2 ! True latitudes of projection  
real :: earth\_radius ! Earth radius, km  
real, dimension(nx,ny) :: slab ! The 2-d array holding the data  
logical :: is\_wind\_grid\_rel ! Flag indicating whether winds are  
! relative to source grid (TRUE) or, U8 V$ o  
! relative to earth (FALSE)  
character (len=8) :: startloc ! Which point in array is given by# s: I, F/ ^/ p) n9 }  
! startlat/startlon; set either& f0 P  F\* J0 L  
! to 'SWCORNER' or 'CENTER 'M  
character (len=9) :: field ! Name of the field  
character (len=24) :: hdate ! Valid date for data YYYY:MM <!--[if !vml]-->file:///C:/DOCUME~1/ADMINI~1/LOCALS~1/Temp/msohtml1/01/clip\_image001.gif<!--[endif]-->D\_HH:00:00  
character (len=25) :: units ! Units of data  
character (len=32) :: map\_source ! Source model / originating center  
character (len=46) :: desc ! Short description of data  
write(unit=ounit) version  
! 2) WRITE METADATA  K( ^\* w( f. Y4 n4 x3 J  
! Cylindrical equidistant  
if (iproj == 0) then  
  
units, desc, xlvl, nx, ny, iproj  
write(unit=ounit) startloc, startlat, startlon, &  
deltalat, deltalon, earth\_radius r0 l) u( Y' p9 ]7 B  
! Mercator  
else if (iproj == 1) then  
units, desc, xlvl, nx, ny, iproj  
write(unit=ounit) startloc, startlat, startlon, dx, dy, &  
truelat1, earth\_radius  
else if (iproj == 3) then% o% n q$ L- E0 ]: U& E9 ]" v6 C# x( L% B  
write(unit=ounit) hdate, xfcst, map\_source, field, &  
write(unit=ounit) startloc, startlat, startlon, dx, dy, &  
0 c; q. o7 o; l( c  
! Gaussian  
write(unit=ounit) hdate, xfcst, map\_source, field, &  
units, desc, xlvl, nx, ny, iproj  
write(unit=ounit) startloc, startlat, startlon, &  
! Polar stereographic  
else if (iproj == 5) then  
! q  {7 b, K/ I9 I  
units, desc, xlvl, nx, ny, iproj  
xlonc, truelat1, earth\_radius  
  
! 3) WRITE WIND ROTATION FLAG  
write(unit=ounit) is\_wind\_grid\_rel  
! 4) WRITE 2-D ARRAY OF DATA  
write(unit=ounit) slab  
生成并编辑Vtables  
尽管已经提供了很多Vtables，它们也对应了很多普遍使用的数据集，但是仍然不可能把所有与ungrib可使用的GRIB格式的数据集的相应 Vtables全部列举。当一个新的源数据需要被ungrib.exe处理时，用户可能需要自己生成一个新的Vtable，或者以某个已经有的 Vtable为模板来修改。无论怎样，对Vtable中各个要素场的意义和用法的基本了解都是有帮助的。每个Vtable可能会包含7个或11个要素场， 这取决于所使用的源数据是GRIB1格式还是GRIB2格式。每个Vtable中的要素场无非是以下三种中的一种：一，描述在GRIB文件中数据如何被确 认的要素场；二，描述数据如何被ungrib.exe和metgrid.exe确认的要素场；三，确定GRIB2格式的要素场。每个被 ungrib.exe提取出的变量在Vtable中都会对应一或更多行，同时多行数据又会根据层次的不同分割—如地面高度和高空层。要素场在Vtable 文件中是对应一行或全部行，取决于场和层次的具体情况。  
第一个要素组—描述在GRIB文件中数据如何被定义的量—的模板如下：

Param| Type |Level1|Level2|  
----- ------ ------ ------   
GRIB1 参数规定了气象要素场的GRIB编码，是在数据集中一系列独一无二的数字。尽管如此，不同的数据集可能会用不同的GRIB编码标示相同的要素场—例 如，GFS数据集中的高层气温的GRIB码是11，但是在ECMWF数据中则是130。为了找到某个要素场的GRIB码，可以使用g1print.exe 和g2print.exe应用程序。在GRIB码中，"Level Type", "From Level1",和"From Level2"是用来确定一个要素场可以在那些层次上找到。与"GRIB1 Param"一样，g1print.exe和g2print.exe可以用来找出多层次要素场的值。多层要素场的意义取决于"Level Type"，总结如下：  
当分层的要素场（Level Type 112）被确定后，层次的起止点是有单位的，这个单位是决定于要素场本身；可以用g1print.exe和g2print.exe应用程序来看这些值。气象,数值,模式,气象数值模式,数值模式,气象模式,海洋模式,气候模式,WRF,MM5,Grapes,HYCOM,LICOM,同化,集合预报,模式联盟,模式论坛0 R/ ]4 ~4 M6 t( E% y  
Vtable的第二个要素场组是来描述在metgrid和real程序中数据如何被确认的，下面是个模板：3 ?( Q& f. x0 K, M( E) [" T) b  
| metgrid | metgrid | metgrid |气  
| Name | Units | Description |  
这些场中最重要的要数 是"metgrid Name"，它决定了变量的名字，这些名字在ungrib写过渡格式文件时被分配到各个气象要素场。这个名字同样要与METGRID.TBL文件中的相匹 配，以便于metgrid程序觉得要素场如何被水平插值。"metgrid Units"和"metgrid Description"分别确定了要素场的单位和一个简单的描述；这里需要特别注意的是，当一个要素场没有被描述时，那ungrib是不会把这个要素场 写进过渡文件的。  
最后一组提供了GRIB2的具体描述，具体如下：  
|GRIB2|GRIB2|GRIB2|GRIB2|  
|Discp|Catgy|Param|Level|  
----------------------- 气  
尽管在一个Vtable文件中有GRIB2要素场并不能妨碍这个Vtable被GRIB1数据使用，而且GRIB2要素场仅仅在需要使用GRIB2数据集 时才会被用到。例如，Vtable.GFS文件包含了GRIB2的要素场，但是却可以被1°（GRIB1）的和0.5°（GRIB2）的GFS数据集使 用。由于WPS提供了很多与GRIB2格式相对于的Vtables，所以在此就不一一列举了。  
用Geogrid二进制格式写静态数据  
geogrid.exe 用来插值的地形静态数据以标准的2-D和3-D数值形式写成一个简单的二进制格式。当用户有了新的静态数据资源可以通过把数据写成二进制形式让WPS来使用。geogrid格式支持单层或多层连续的场，场的分类（作为区域的分类）及为每种分类给出的一小部分的场。根据二进制格式的含义，下面给出了一个 30-second的USGS土地利用数据集及的模板：  
  
对 于一个主导分类的分类要素场，数据首先要存储在一个规则的2-D整型数组中，每个整数都给出了与格点相对应的主导分类的说明。考虑到这个数组的作用，数据 是由低（空）到高（空）或由南向北一行行的写入文件中。例如，在上边的图中，n×m的数组的各个元素是按照x11, x12, ..., x1m, x21, ... , x2m, ... , xn1, ... , xnm的顺序写入的。尽管可以通过调整“index”文件中的参数endian=little来设置数据集中排列顺序为little-endian，在写 入文件时，每个元素是按big-endian的顺序排列的（例如，对于4-字节整数ABCD，字节A被存储在最低的地址中，而字节D被存储在最高的字节 中）。而且虽然在一个数组中用所需的最小数字来代表一个完整的数值范围是最好的，但在这里一个文件中必须用相同的为位数来存储各个元素。当写入二进制数据 时，不用头文件，记录标志，或者多余的位数。例如，一个两位的1000×1000数组文件的大小是2,000,000。因为Fortran无格式的写记录 标志，因此它是不可能直接写出一个geogrid的二进制文件的，但是可以通过用C或Fortran在写或读时调用read\_geogrid.c和 write\_geogrid.c（在geogrid/src目录下）来实现。与主导分类类似的是连续或真实的要素场。像主导分类一样，单层连续场作为2- D数组首先被组织，然后一行行写入二进制文件。尽管如此，因为一个连续长可以包含非整型或负值，所以每个文件中的存储表示有点复杂。数组中所有的元素都要 先被转成整型。首先通过扫描所有的元素，找出有需要的精度的，然后把剩下的部分通过四舍五入来移除。例如，要求精度达到小数点后三位，则-2.71828 需要被舍入成-2.718。然后所有的的数组元素都被转化成整数，如果有负数，则会有第二次转化：如果用一个字节储存一个元素，则每个负数都会添加28； 如果是两个字节则是216；三个字节是224；四个字节是232。这对无需转化成正数的过程很重要。最后的正的整数数组按照主导分类的格式被写入文件。多 层的联系场类似单层的。对于一个n × m × r数组，先按照上边的方法转化成正的整数场，然后从低到高开始写每个层次。分类场可以被看成是一个多层连续场的一部分，它的层次是k（1 ≤ k ≤ r）。当把一个场按照geogrid二进制格式写入一个文件时，用户应该遵守geogrid程序的命名规则，即希望数据文件的名字格式是xstart- xend.ystart-yend，其中xstart, xend, ystart,和yend分别是5位正整数，数值的起止x指针、起止y指针都包含在文件中；指针起始为1，而不是0。所以，一个800×1200的数组 （如800行，1200列）可以被命名为00001-01200.00001-00800。当一个数据被分成多个部分，每个部分都是一个规则的矩形数组， 每个数组都是一个单独的文件。在这种情况下，数组相对的位置是由每个数组的文件名中的x和y指针决定的。尽管如此，值得注意的是，数据集中的每个部分都要 有相同的x-和y-维数，并且每一块数据不能相互重叠；并且每块的起止都要在索引的范围内。例如，全球的30-second的USGS地形数据集被分成多个1200×1200的数组，每个数组都代表了一个10°×10°的区域；这个文件的最西南角的位置是（90S, 180W），命名为00001-01200.00001-01200，而最东北角的位置是（90N, 180E），命名为42001-43200.20401-21600。如果数据集被分成多个小的部分，而且格点的数字不能在x-方向上被平分，那最后剩下 的几列数据就要用标志数（index file文件中参数missing\_value确定）来填补。例如，一个x-方向上有2456个格点的数据集，如果分成3个部分的话，在规定范围内每块应 该是1–820,821–1640,和1641–2460，而2457-2460之间就要用标志数（类似缺测值）来填充。  
因为起 止索引必须是5位数字，所以一个场是不能在x-或y-方向上超过99999个数据格点的，如果超过了，用户要把这个数据集分成多个geogrid能识别的 小的部分。例如，一个大的全球数据集可以把东半球和西半球分成多个小的部分。除了二进制数据文件，geogrid要求每一个数据集都有一个额外的说明文 件，即“index”文件，因此，两个数据集是不能放在同一个目录下的。其实，当要处理数据集时，index文件是最先被geogrid检索的，因为这个 文件里包含了所有可能与用到的数据相关的有用信息，下面是一个index的模板：  
type = continuous  
signed = yes  
projection = regular\_ll  
dx = 0.00833333  
known\_x = 1.0  
known\_y = 1.0  
known\_lat = -89.99583  
known\_lon = -179.99583  
Mtile\_x = 1200  
tile\_z = 1  
tile\_bdr=3  
description="Topography height"  
用户可以通过参考index file options来了解所有可能出现在index文件中的参数及相关的意义。