



**MINISTÉRIO DA DEFESA
COMANDO DA AERONÁUTICA
INSTITUTO DE CONTROLE DO ESPAÇO AÉREO**

**MANUAL DE INSTALAÇÃO, COMPILAÇÃO E EXECUÇÃO DO
SISTEMA DE MODELAGEM NUMÉRICA WRF NO ICEA.**

**Paulo Geovani Iriart Bolsista DTI (ACA/IAE)
Miguel Ângelo Vargas de Carvalho Cap Esp Met (PBCA/ICEA)
Antônio Vicente Pereira Neto Cap Esp Met (CNMT/CNMA)**

Projeto de pesquisa coordenado pelo Grupo de Trabalho do Programa de Modelagem
Numérica de Tempo (PMNT)

São José dos Campos – SP
PBCA/ICEA
2011

SUMÁRIO

	Pág.
1 INTRODUÇÃO.....	5
1.1 Objetivo	6
2 DESCRIÇÃO DO MODELO WRF	7
3 INSTALAÇÃO, COMPILAÇÃO E EXECUÇÃO DOS SUBSISTEMAS DO WRF	8
3.1 Preparação inicial.....	9
3.2 Bibliotecas opcionais para o GRIB2.....	10
3.2 Instalação do NetCDF.....	11
3.3 Instalação do WRF.....	12
3.1.1 Instalação do subsistema WRFV3	13
3.1.2 Instalação do subsistema WPS.....	14
3.1.3 Instalação do subsistema ARWpost.....	16
3.4 Dados geográficos e de terreno.....	17
3.5 Instalação do pacote gráfico GrADS	18
4 CONFIGURAÇÕES INICIAIS DO WRF	19
5 EXECUTANDO O WRF	23
5.1. Execução do WPS.....	24
5.1.1 Execução GEOGRID	24
5.1.2 Execução UNGRIB	26
5.1.1 Execução METGRID	27
5.2 Execução do Real e WRF.....	29
5.3 Execução do ARWpost	31
6 CONSIDERAÇÕES FINAIS.....	33
7 AGRADECIMENTOS.....	34
REFERÊNCIAS BIBLIOGRÁFICAS	34
APÊNDICE A – NAMELISTS GRADE SUDESTE.....	36
APÊNDICE B – NAMELISTS GRADE NORDESTE.....	40
APÊNDICE C – NAMELISTS GRADE NORTE.....	44
APÊNDICE D – NAMELISTS GRADE ALCÂNTARA.....	48

1. INTRODUÇÃO

Os modelos de circulação geral da atmosfera (MCGA) e os modelos regionais (MREG) da atmosfera são hoje as ferramentas mais importantes para se realizar previsões numéricas de tempo (PNT). Em suma, um modelo de PNT é um sistema desenvolvido em computador para simular o comportamento da atmosfera. Isto é possível, porque ele resolve um conjunto de equações matemáticas baseadas nas leis físicas, aplicadas para atmosfera (fluido geofísico), de modo a prever o estado futuro da atmosfera partindo de condições iniciais específicas. Segundo Tomé (2004), os modelos de PNT podem ser classificados segundo a sua escala espacial:

- Global, com resolução da ordem de 200 km, visa identificar o comportamento geral da atmosfera sobre uma área extensa, estes modelos são apenas capazes de identificar fenômenos meteorológicos de larga escala (chamados de sinóticos);
- Regional, com resolução espacial de alguns metros a cerca de 50 km, visa identificar com maior detalhe o comportamento da atmosfera sobre uma região específica, sendo capazes de identificar fenômenos meteorológicos de média escala (chamados de mesoscala).

Os modelos regionais (MREG) permitem resoluções mais altas que os modelos globais e podem ser utilizados aplicando-se *downscaling* nas previsões dos modelos globais e/ou análises (CHOU, 2002). O propósito do *downscaling* é obter detalhes regionais ou locais a partir de observações ou simulações numéricas de baixa resolução. Uma das vantagens da utilização de resoluções mais altas nos MREG é a possibilidade de melhor representação dos fenômenos meteorológicos de escalas espaciais e temporais menores, dos processos topográficos e das características da superfície.

Para o Comando da Aeronáutica, a previsão de fenômenos meteorológicos é importante por causa do impacto nas atividades aeronáuticas e aeroespaciais, tais como gerenciamento do tráfego aéreo e lançamento de foguetes. Em 2002, foram iniciadas, tanto para fins operacionais quanto para pesquisa e desenvolvimento científico, as primeiras simulações com a 5ª Geração do Modelo de Mesoescala (MM5), desenvolvido conjuntamente pela *Pennsylvania State University* (PSU) e pelo *National*

Center for Atmospheric Research (NCAR). Atualmente, o MM5 tem sido usado pelas seguintes instituições: Instituto de Aeronáutica e Espaço (IAE), Instituto de Controle do Espaço Aéreo (ICEA) e Centro Nacional de Meteorologia Aeronáutica (CNMA). Os prognósticos gerados pelo MM5 estão disponibilizados no sítio do ICEA¹ e da Rede de Meteorologia do Comando da Aeronáutica (REDEMET)². Essas Organizações Militares em parceria com outras Instituições Acadêmicas têm desenvolvido pesquisas com o MM5, como por exemplo: Oyama (2003; 2006), Barros e Oyama (2010), Pereira Neto (2010), Nogueira (2010), Correa et al. (2010), Carvalho (2011), entre outros.

O MM5 é um modelo muito robusto e consistente especialmente pela sua maturidade, i.e., por incluir mais de duas décadas de melhoria. Maiores detalhes sobre o MM5 podem ser encontrados no sítio <http://box.mmm.ucar.edu/mm5>. Entretanto, desde 2003, o desenvolvimento do MM5 foi descontinuado pela PSU/NCAR e, portanto, novas gerações do sistema de modelagem não estão previstas. O grupo de desenvolvimento do MM5 passou a implementar um novo sistema de modelagem em cooperação com diversas instituições dos Estados Unidos da América. Iniciou-se a elaboração e desenvolvimento do *Weather Research and Forecasting Model* (WRF). Com os esforços direcionados para o desenvolvimento do WRF, a sua utilização tem crescido. No Brasil, por exemplo, é utilizado em pesquisas das mais conceituadas Instituições Acadêmicas, entre elas Instituto Nacional de Pesquisas Espaciais (INPE), Universidade Federal do Rio de Janeiro (UFRJ) e Universidade de São Paulo (USP). No âmbito do Comando da Aeronáutica, por intermédio seu Boletim (BCA) nº 196, de 15 de outubro de 2008, foi constituído o Grupo de Trabalho (GT) do Programa de Modelagem Numérica do Tempo (PMNT). O GT, com o intuito de acompanhar os novos avanços na área de previsão numérica de tempo, escolheu o WRF como o próximo modelo a ser implementado no ICEA.

1.1. Objetivo

O objetivo deste trabalho é descrever as etapas de instalação, compilação e execução do sistema de modelagem numérica WRF, projeto desenvolvido no ICEA em parceria com o IAE e o CNMA.

¹ www.icea.gov.br/climatologia/produtos

² www.redemet.aer.mil.br

2. DESCRIÇÃO DO MODELO WRF

O WRF é a última geração de modelo numérico de previsão do tempo que servirá tanto para a operacionalidade dos centros meteorológicos como para as pesquisas atmosféricas. Ele foi desenvolvido através de uma parceria entre diversos órgãos, entre eles o *National Center for Atmospheric Research* (NCAR), *National Oceanic and Atmospheric Administration* (NOAA), *National Center for Environmental Prediction* (NCEP), *Forecast Systems Laboratory* (FSL), *Air Force Weather Agency* (FAWA), e outros órgãos de pesquisa e desenvolvimento. Maiores informações sobre o WRF podem ser obtidas no sítio <http://www.wrf-model.org>.

As características do modelo que mais se destacam são: os múltiplos núcleos dinâmicos, sistema variável de assimilação de dados tri-dimensional e uma estrutura de software que permite o paralelismo computacional, bem como a extensibilidade do sistema. Pode ser instalado em diversas plataformas computacionais (p. ex. Linux), é muito portátil, flexível, de domínio público e disponibilizado gratuitamente pela internet. O modelo pode ser executado tanto para situações atmosféricas idealizadas como situações reais, em um espectro amplo de aplicações em escalas horizontais que variam de milhares de quilômetros a poucos metros.

O sistema de modelagem do WRF compreende diversos componentes (Fig. 1; para maiores detalhes, veja *ARW V3 Modeling System User's Guide*, 2011). Os principais são: no pré processamento o WRF Preprocessing System (WPS); inicialização do WRF (real); execução do WRF; e pós-processamento (ARWpost). O WPS é um conjunto de três subsistemas que prepara os dados de entrada para a simulação: geogrid, ungrrib e metgrib. O geogrid define o domínio do modelo e interpola os dados terrestres para as grades. O ungrrib extrai os campos meteorológicos do formato GRIB (V1 e V2) necessários para inicializar o modelo. O metgrib interpola horizontalmente os campos meteorológicos extraídos pelo ungrrib para as grades do modelo definida pelo geogrid. O trabalho de interpolar verticalmente os campos do WRF é feito dentro do programa real. O WRF é responsável pela execução do modelo propriamente dito. Por fim, o ARWpost é usado para converter as saídas do modelos para um formato que possam ser visualizados por aplicativos gráficos.

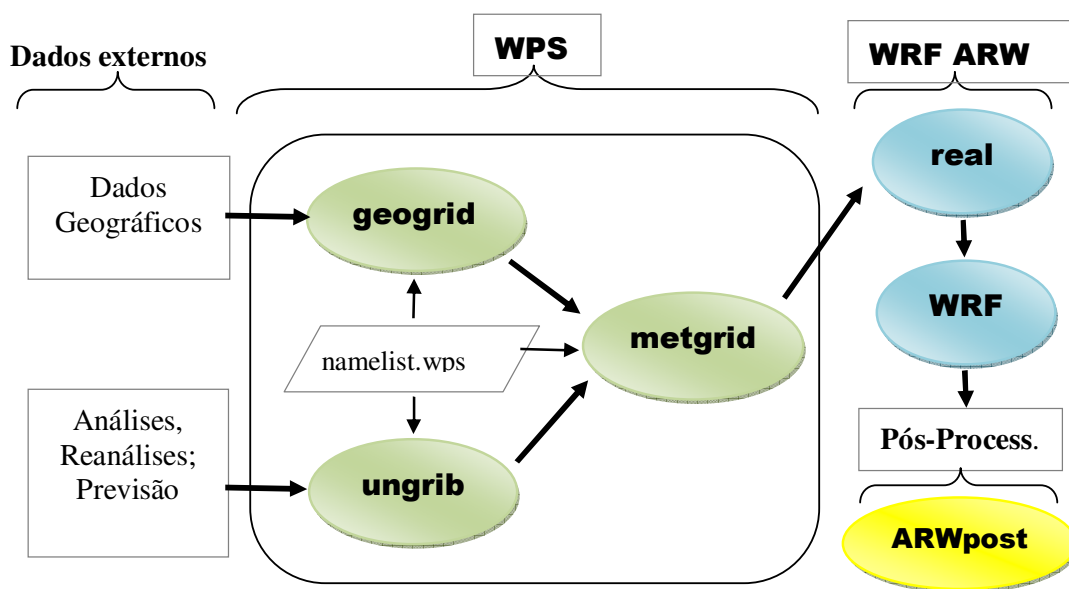


Figura 1 – Representação esquemática do sistema de modelagem. Adaptado de ARW V3 Modeling System User’s Guide, 2011.

3. INSTALAÇÃO, COMPILAÇÃO E EXECUÇÃO DOS SUBSISTEMAS DO WRF

O modelo WRF versão 3.2.1 foi instalado sob o sistema operacional livre Debian³ versão 5.0 “Lenny” de 32bits. O multiprocessamento, que possibilita diversos fluxos de execução simultâneos, foi escolhido para a execução do modelo. O WRF foi desenvolvido para que explore eficientemente o potencial do uso do paralelismo computacional.

Os principais componentes do *hardware* são: quatro servidores da HP Proliant (DL360 G7), com 01 (um) processador Quad-Core com clock de 2.4 GHz (por core) com 6 (seis) GB de memória RAM; 2 a 4 unidades de disco rígido internos de 146 GB; e memória cachê de 256MB. Cada servidor simulará um domínio. Maiores detalhes da configuração dos domínios serão fornecidos mais adiante.

Para que o WRF funcione corretamente, é necessária a instalação de diversos aplicativos (bibliotecas, compiladores, linguagens de programação, etc.), bem com do NetCDF e dos arquivos de instalação dos subsistemas do modelo. Alguns aplicativos estão disponíveis no repositório de *software* do Linux, outros em sítios da *internet*. Para procurar, instalar ou manipular pacotes de *software* a partir dos repositórios do Linux

³ www.debian.org/releases/lenny/index.en.html

utiliza-se a ferramenta *Advanced Packaging Tool* (APT). Os arquivos da *internet* podem ser baixados diretamente para um diretório a sua escolha (neste trabalho, /home/WRF/install/). Os sítios necessários para instalação do modelo serão informados neste trabalho. A seguir, será apresentada uma descrição detalhada de cada etapa da implementação do modelo.

3.1 Preparação Inicial

Antes da instalação dos subsistemas do WRF, é necessário preparar os servidores para execução do modelo. Para isto, são instalados ou atualizados diversos aplicativos (bibliotecas do Linux, compiladores C/C++ e Fortran, linguagens de programação, etc.). Com a ferramenta APT, são instalados os seguintes aplicativos: gfortran, gcc, g++, cpp, flex, csh, libncurses5-dev, bison, curl, m4, perl, make, zip. Como a execução do modelo ocorre em paralelo, ainda são instalados: Openmpi-bin, Openmpi-doc, Libopenmpi-dev. Todos os programas devem ser instalados antes da instalação dos subsistemas do WRF. Os comandos no terminal do Linux são executados como root do sistema.

```
# apt-get update
# apt-get install gfortran
# apt-get install gcc
# apt-get install g++
# apt-get install cpp
# apt-get install flex
# apt-get install csh
# apt-get install bison
# apt-get install curl
# apt-get install m4
# apt-get install perl
# apt-get install make
# apt-get install zip
# apt-get install libncurses5-dev
# apt-get install openmpi-bin
# apt-get install openmpi-doc
# apt-get install libopenmpi-dev
```

3.2 Bibliotecas opcionais para GRIB2

As condições iniciais e de contorno utilizadas no modelo estão no formato da segunda versão do código *General Regularly-distributed Information in Binary* (GRIB2). O código GRIB é um formato de arquivo padrão da *World Meteorological Organization* (WMO) para o armazenamento de campos regularmente distribuídos (p.ex., pontos de grade). Para a extração dos campos meteorológicos desses arquivos pelo WRF, é necessária a instalação de três bibliotecas: JasPer 1.900.1⁴, Zlib 1.2.5⁵ e PNG 1.4.7⁶. As bibliotecas são, então, copiadas e descompactadas no diretório /usr/local.

```
# cp -f /home/WRF/install/jasPer-1.900.1.zip libpng-1.4.7.tar.gz zlib-1.2.5.tar.gz /usr/local/
# cd /usr/local
# unzip jasPer-1.900.1.zip
# tar -vzxf libpng-1.4.7.tar.gz
# tar -vzxf zlib-1.2.5.tar.gz
```

Concluída a descompactação dos arquivos, deve-se entrar nos três diretórios criados (Fig.2, zlib-1.2.5, jasPer-1.900.1 e libpng-1.4.7) e compilar as respectivas bibliotecas:

```
# cd zlib-1.2.5
# ./configure --prefix=/usr/local
# make
# make intall
# cd ../jasper-1.900.1
# ./configure --prefix=/usr/local
# make
# make intall
# cd ../libpng-1.4.7
# ./configure --prefix=/usr/local
# make check
# make intall
# cd ..
```

⁴ www.ece.uvic.ca/~mdadams/jasper/

⁵ www.zlib.net

⁶ www.libpng.org/pub/png/libpng.html

O *script* de instalação do WPS possui versões diferentes dessas bibliotecas. Mais adiante, será mostrado como alterar a versão das bibliotecas que estão localizadas dentro do arquivo `configure.wps`.

3.3 Instalação do NetCDF

A versão do NetCDF instalada é a 3.6.3 (não compilada), disponibilizada por *Unidata Program Center*⁷ (“NetCDF C/C++/Fortran Stable Releases”). Após o *download*, o arquivo é copiado e descompactado para o diretório `/usr/local`:

```
# cp -f /home/WRF/install/netcdf-3.6.3.tar.gz /usr/local/  
# cd /usr/local/  
# tar -vzxf netcdf-3.6.3.tar.gz
```

O diretório criado (`netcdf-3.6.3`) pode ser visualizado na Figura 2. Entre nesse diretório, configure e compile o NetCDF. Uma mensagem, no final da instalação, informará o êxito da instalação.

```
# cd netcdf-3.6.3  
# ./configure --prefix=/usr/local/netcdf  
# make  
# make check  
# make install  
# cd ..  
“Congratulations! You have successfully installed netCDF”
```

Se ocorrerem erros, informações adicionais podem ser obtidas no Guia de Instalação do NetCDF⁸.

⁷ www.unidata.ucar.edu/downloads/netcdf/index.jsp

⁸ www.unidata.ucar.edu/software/netcdf/docs/netcdf-install

```
webpca@pesquisa: /usr/local$ ls
bin      jasper-1.900.1      man          share
bw       jasper-1.900.1.zip netcdf       src
etc      lib                  netcdf-3.6.3  zlib-1.2.5
games    libpng-1.4.7        netcdf-3.6.3.tar.gz  zlib-1.2.5.tar.gz
include  libpng-1.4.7.tar.gz sbin
webpca@pesquisa: /usr/local$ █
```

Figura 2 – Figura ilustrativa do terminal mostrando a pasta /usr/local. Os retângulos vermelhos mostram os diretórios criados após a descompactação das bibliotecas. O retângulo amarelo é o diretório criado após a instalação do NetCDF.

3.4 Instalação do WRF

Os arquivos necessários para instalação dos subsistemas do WRF estão disponibilizados no sítio *University Corporation for Atmospheric Research (UCAR)*⁹. O local escolhido para instalação do modelo é o diretório /home/WRF. Inicialmente, são baixados e transferidos para o novo diretório os arquivos de instalação do WRF (versão 3.2.1), WPS (versão 3.1.1) e ARWPost (versão 2.2), bem como os dados geográficos e de terreno para a montagem dos domínios.

```
# mkdir /home/WRF
# cp -f /home/WRF/install/geog_v3.1.tar.gz WPSV3.1.1.TAR WRFV3.2.1.TAR
ARWpost.tar.gz /home/WRF
# cd /home/WRF
```

Para todos os subsistemas do modelo, os procedimentos para instalação são semelhantes. Após descompactar o arquivo, o NetCDF é definido como variável ambiental e para que ele possa suportar arquivos grandes (>2GB). O próximo passo é realizar a configuração e compilação dos subsistemas. A sequência de comandos utilizados para a instalação de cada subsistema do WRF é mostrada abaixo:

⁹ http://www.mmm.ucar.edu/wrf/users/download/get_source.html

3.4.1 Instalação do subsistema WRFV3:

```
# tar -vxf WRFV3.2.1.TAR
# cd WRFV3
# export NETCDF= /usr/local/netcdf      ! (setar a variável de ambiente)
# export WRFIO_NCD_LARGE_FILE_SUPPORT=1 ! (arquivos grandes)
# ./configure                          ! (escolher a opção dentre as oferecidas, Fig. 3)
# ./compile em_real >& compile.log
# cd ..
```

O *script* configure cria o arquivo de configuração do WRF. Sua função é checar os recursos computacionais (*hardware* e *software*) e oferecer aos usuários opções para a configuração do WRF. O arquivo criado chama-se *configure.wrf*. Este novo arquivo também pode ser editado com outras opções, caso seja necessário. Para a compilação do WRF, utiliza-se o *script* compile. Se for executado com sucesso, os seguintes executáveis são criados no diretório /main: *ndown.exe*, *tc.exe*, *nup.exe*, *real.exe* e *wrf.exe*. Caso um ou mais destes arquivos não tenham sido criados, procurar erros dentro do arquivo *compile.log*, corrigi-los e repetir a sequência de comando deste tópico. Antes de recompilar, execute o *script* clean (*./clean -a*). A Figura 3 mostra as opções escolhidas para a instalação do modelo nos servidores do ICEA e a Figura 4, os arquivos criados na pasta WRFV3.

```
-----
Please select from among the following supported platforms.
1. Linux i486 i586 i686, gfortran compiler with gcc (serial)
2. Linux i486 i586 i686, gfortran compiler with gcc (smpar)
3. Linux i486 i586 i686, gfortran compiler with gcc (dmpar)
4. Linux i486 i586 i686, gfortran compiler with gcc (dm+sm)
5. Linux i486 i586 i686, g95 compiler with gcc (serial)
6. Linux i486 i586 i686, g95 compiler with gcc (dmpar)
7. Linux i486 i586 i686, PGI compiler with gcc (serial)
8. Linux i486 i586 i686, PGI compiler with gcc (smpar)
9. Linux i486 i586 i686, PGI compiler with gcc (dmpar)
10. Linux i486 i586 i686, PGI compiler with gcc (dm+sm)
11. Linux x86_64 i486 i586 i686, ifort compiler with icc (serial)
12. Linux x86_64 i486 i586 i686, ifort compiler with icc (smpar)
13. Linux x86_64 i486 i586 i686, ifort compiler with icc (dmpar)
14. Linux x86_64 i486 i586 i686, ifort compiler with icc (dm+sm)
15. Linux i486 i586 i686 x86_64, PathScale compiler with pathcc (serial)
16. Linux i486 i586 i686 x86_64, PathScale compiler with pathcc (dmpar)

Enter selection [1-16]: 2
-----
Compile for nesting? (0=no nesting, 1=basic, 2=preset moves, 3=vortex following)
[default 0]: 1
```

Figura 3 – Figura ilustrativa do terminal mostrando as opções escolhidas para a compilação do WRF. Os retângulos vermelhos mostram as opções selecionadas.

```

webpca@pesquisa:/home/WRF/WRFV3/main$ ls
convert_em.F          module_wrf_top.o    nup.exe             tc_em.o
ideal.F              ndown_em.F         real_em.F          tc.exe
libwrflib.a          ndown_em.f90      real_em.f90       wrf_ESM6_wrf.F
Makefile             ndown_em.o        real_em.o         wrf.exe
module_initialize_real.mod ndown.exe         real.exe          wrf.F
module_wrf_top.F     nup_em.F          real_nmmr.F       wrf.f90
module_wrf_top.f90  nup_em.f90       tc_em.F           wrf.o
module_wrf_top.mod  nup_em.o         tc_em.f90        wrf_SST_ESMF.F
webpca@pesquisa:/home/WRF/WRFV3/main$

```

Figura 4 – Figura ilustrativa do terminal mostrando a pasta /home/WRF/WRFV3/main. Os retângulos vermelhos mostram os executáveis criados após a compilação do WRF.

Obs.: A compilação do WRF leva algum tempo, pois todo o código é compilado. Caso não ocorram alterações, a compilação é realizada somente uma vez.

3.4.2 Instalação do subsistema WPS:

```

# tar -vxf WPSV3.1.1.TAR
# cd WPS
# export NETCDF= /usr/local/netcdf
# export WRFIO_NCD_LARGE_FILE_SUPPORT=1
# ./configure                ! (escolher a opção dentre as oferecidas, Fig. 5)
# ./compile >& compile.log
# cd ..

```

Como informado anteriormente, a sequência de comandos utilizada para configuração e compilação do WPS é semelhante à mostrada para o WRF. Agora, o arquivo de configuração criado é o `configure.wps` e, após a compilação do WPS, os seguintes executáveis são criados nos diretórios: `geogrid.exe` (define o tamanho e localização do domínio) em `geogrid/src`; `ungrib.exe` (extrai os campos meteorológicos dos arquivos GRIB) em `ungrib/src`; e `metgrid.exe` (interpola horizontalmente os campos meteorológicos para a grade simulada definida) em `metgrid/src`. Os atalhos desses executáveis são criados no diretório `/WPS`. Caso um ou mais desses arquivos não tenham sido criados ou o nome deles estiver piscando, procurar erros dentro do arquivo `compile.log`, corrigi-los e repetir a sequência de comandos deste tópico. Antes de recompilar, execute o *script* `clean` (`./clean -a`). A Figura 5 mostra as opções escolhidas para a instalação do modelo nos servidores do ICEA e a Figura 6, os atalhos criados no diretório `/WPS`.

```

Will use NETCDF in dir: /usr/local/netcdf
$JASPERLIB or $JASPERINC not found in environment, configuring to build without
grib2 I/O...
-----
Please select from among the following supported platforms.

  1. PC Linux i486 i586 i686, PGI compiler      serial, NO GRIB2
  2. PC Linux i486 i586 i686, PGI compiler      serial
  3. PC Linux i486 i586 i686, PGI compiler      DM parallel, NO GRIB2
  4. PC Linux i486 i586 i686, PGI compiler      DM parallel
  5. PC Linux i486 i586 i686, Intel compiler     serial, NO GRIB2
  6. PC Linux i486 i586 i686, Intel compiler     serial
  7. PC Linux i486 i586 i686, Intel compiler     DM parallel, NO GRIB2
  8. PC Linux i486 i586 i686, Intel compiler     DM parallel
  9. PC Linux i486 i586 i686, g95 compiler,     serial, NO GRIB2
 10. PC Linux i486 i586 i686, g95 compiler,     serial
 11. PC Linux i486 i586 i686, g95 compiler,     DM PARALLEL, NO GRIB2
 12. PC Linux i486 i586 i686, g95 compiler,     DM PARALLEL
 13. PC Linux i486 i586 i686, gfortran compiler, serial, NO GRIB2
 14. PC Linux i486 i586 i686, gfortran compiler, serial
 15. PC Linux i486 i586 i686, gfortran compiler, DM PARALLEL, NO GRIB2
 16. PC Linux i486 i586 i686, gfortran compiler, DM PARALLEL

Enter selection [1-16] : 14

```

Figura 5 – Figura ilustrativa do terminal mostrando as opções escolhidas para a compilação do WPS. Os retângulos vermelhos mostram as opções selecionadas.

Obs.: A execução do WPS ocorre em série, pois o ganho da execução deste subsistema em paralelo é muito pequeno.

```

FILE:2010-03-22_18 met_em.d02.2010-03-21_12:00:00.nc
FILE:2010-03-23_00 met_em.d02.2010-03-21_18:00:00.nc
FILE:2010-03-23_06 met_em.d02.2010-03-22_00:00:00.nc
FILE:2010-03-23_12 met_em.d02.2010-03-22_06:00:00.nc
geo_em.d01.nc met_em.d02.2010-03-22_12:00:00.nc
geo_em.d02.nc met_em.d02.2010-03-22_18:00:00.nc
geogrid met_em.d02.2010-03-23_00:00:00.nc
geogrid.exe met_em.d02.2010-03-23_06:00:00.nc
geogrid.tog met_em.d02.2010-03-23_12:00:00.nc
GRIBFILE.AAA metgrid
GRIBFILE.AAB metgrid.exe
GRIBFILE.AAC metgrid.tog
GRIBFILE.AAD namelist.wps
GRIBFILE.AAE namelist.wps.all_options
GRIBFILE.AAF namelist.wps.fire
GRIBFILE.AAG namelist.wps.global
GRIBFILE.AAH namelist.wps.nmm
GRIBFILE.AAI README
GRIBFILE.AAJ ungrib
GRIBFILE.AAK ungrib.exe
GRIBFILE.AAL ungrib.tog
GRIBFILE.AAM util
link_grib.csh vtable
webpca@pesquisa: /home/WRF/WPS$

```

Figura 6 – Figura ilustrativa do terminal mostrando a pasta /home/WRF/WPS. Os retângulos vermelhos mostram os executáveis criados após a compilação do WPS.

Erros podem ocorrer associados à ausência das variáveis ambientais. Para checar se elas estão criadas no diretório WPS, digite o seguinte comando no terminal do Linux:

```
# env | grep -i NETCDF
```

Durante a instalação do WPS, pode ser necessário mudar a versão da biblioteca libpng, indicada na variável `COMPRESSION_LIBS` que está localizada dentro do arquivo `configure.wps` (somente o texto em **negrito** foi modificado).

```
# nano configure.wps
(...)
COMPRESSION_LIBS = -L/data3a/mp/gill/WPS_LIBS/local/lib \
                   -ljasper -lpng14 -lpng -lz
```

Os erros mais comuns que ocorrem nesta operação estão associados à versão incorreta das bibliotecas do GRIB2. Normalmente, o `ungrib.exe` não é criado após a compilação. Uma dica, para verificar se os erros de compilação estão associados às bibliotecas do GRIB2, é realizar a compilação com as opções NO GRIB2 (p.ex.: opção 13, Fig. 5). Outra dica, é a utilização do comando `ldd` para determinar quais são as bibliotecas necessárias de um executável. (p.ex.: `ldd ungrub.exe`). O comando `ldd` – List Dynamic Dependencies – fornece uma lista das dependências dinâmicas de que um determinado programa precisa para o seu correto funcionamento. Normalmente, ele retorna o nome de uma ou mais bibliotecas compartilhadas e suas localizações esperadas. Caso elas estejam ausentes, elas podem ser transferidas para o local esperado (p.ex.: `/usr/lib`).

3.4.3 Instalação do subsistema ARWpost:

```
# tar -vzxf ARWpost.tar.gz
# cd ARWpost
# export NETCDF= /usr/local/netcdf
# export WRFIO_NCD_LARGE_FILE_SUPPORT=1
# ./configure                ! (escolher a opção dentre as oferecidas, Fig. 8)
# ./compile >& compile.log
# cd ..
```

No último subsistema, após a sua configuração e compilação, o arquivo `ARWpost.exe` deve ter sido criado. Caso isto não ocorra, procurar erros dentro do arquivo `compile.log`, corrigi-los e repetir a sequência de comandos deste tópico. Antes de recompilar, execute o *script* `clean` (`./clean-a`). A Figura 7 mostra as opções escolhidas para a instalação do modelo nos servidores do ICEA e a Figura 8, os arquivos criados no diretório `WRFV3`.


```
webpca@pesquisa:/home/WRF/ARWpost$ ./configure
Will use NETCDF in dir: /usr/local/netcdf
-----
Please select from among the following supported platforms.

  1. PC Linux i486 i586 i686 x86_64, PGI compiler (no vis5d)
  2. PC Linux i486 i586 i686 x86_64, PGI compiler (vis5d)
  3. PC Linux i486 i586 i686 x86_64, Intel compiler (no vis5d)
  4. PC Linux i486 i586 i686 x86_64, Intel compiler (vis5d)
  5. PC Linux i486 i586 i686 x86_64, gfortran compiler (no vis5d)
  6. PC Linux i486 i586 i686 x86_64, gfortran compiler (vis5d)

Enter selection [1-6] : 5
```

Figura 7 – Figura ilustrativa do terminal mostrando as opções escolhidas para a compilação do ARWpost. Os retângulos vermelhos mostram as opções selecionadas.

```
webpca@pesquisa:/home/WRF/ARWpost$ ls
arch          configure.arwp  namelist.ARWpost  test.dat
ARWpost.exe  fields.plt     README            util
clean        gribinfo.txt   scripts           wrfout_d01_2010-03-20_12:00:00
compile      gribmap.txt    src               wrfout_d02_2010-03-20_12:00:00
configure    myLIST         test.ctl
webpca@pesquisa:/home/WRF/ARWpost$
```

Figura 8 – Figura ilustrativa do terminal mostrando a pasta /home/WRF/ARWpost. O retângulo vermelho mostra o executável criado após a compilação do ARWpost.

3.4 Dados geográficos e de terreno:

Os dados geográficos e de terreno são descompactados no diretório data, criado dentro de /home/WRF. Esses dados são obtidos dentro de um único arquivo (geog.tar.gz) que, após descompactado, cria um novo diretório (geog) com aproximadamente 11GB.

```
# mkdir /home/WRF/data
# cd /home/WRF/data
# tar -vzxf geog.tar.gz
```

Obs.: Mais adiante, na execução do WPS, será editado o arquivo namelist.wps (variável geog_data_path) para indicar o caminho dos dados geográficos.

O conjunto de dados da superfície terrestre pertencente ao arquivo geog.tar.gz (dados estáticos) foram obtidos através de mapeamento por satélite. Parte desse arquivo contém categorias compatíveis com as da *United States Geological Survey* (USGS) que estão

descritos nos arquivos VEGPARM.TBL e SOILPARM.TBL, localizados no diretório run do WRF. As resoluções dos dados são variadas (1°, 10', 5', 2' e 30'') e contêm as seguintes informações: albedo da superfície mensal (albedo_ncep); fração da vegetação mensal (greenfrac), índice de inclinação (slope index); categoria de uso da terra (landuse); albedo máximo da neve (maxsnowalb); dados de superfície (modis_landuse_20class_30s; apenas para usuários de modelagem de superfície da Noah LSM); dados para os esquemas de ondas de gravidade (orogwd); média anual de temperatura do solo de acordo com a profundidade (soiltemp); tipo de solo da camada inferior (soiltype bot); tipo de solo da camada superior (soiltype top); altura da topografia (topo).

3.5 Instalação do pacote gráfico GrADS:

O *Grid Analysis and Display System* (GrADS) é uma das ferramentas gráficas mais difundidas e utilizadas no mundo para facilitar o acesso, manipulação e visualização de dados das principais variáveis atmosféricas e terrestres. O GrADS é distribuído gratuitamente na *internet* e suporta muitos formatos de arquivo de dados, incluindo o GRIB (versão 2), utilizado como condições iniciais e de contorno no modelo instalado no ICEA.

Primeiramente, o código fonte da versão 1.9 do GrADS (grads-1.9b4-linuxRHE3.tar.gz) é copiado e descompactado em /usr/local. Os códigos fontes pré-compilados estão disponibilizados no sítio do GrADS¹⁰. Os arquivos executáveis (gradsc, gradenc, gribmap, etc) são copiados para o diretório /usr/local/bin. O próximo passo é criar o diretório grads em /usr/local/lib e copiar os arquivos das bibliotecas do GrADS, referentes aos arquivos de mapas e fontes.

```
# cp -f /home/WRF/install/grads-1.9b4-linuxRHE3.tar /usr/local
# tar -vzxf grads-1.9b4-linuxRHE3.tar.gz
# cd grads-1.9b4/bin
# cp -f * /usr/local/bin
# mkdir /usr/local/lib/grads
# cp -fr grads-19b4/data/* /usr/local/lib/grads
```

¹⁰ <http://grads.iges.org/grads/downloads.html>

A variável ambiental GASCRP é criada para indicar ao sistema operacional e demais aplicativos a localização da lista de diretórios contendo *scripts* do GrADS e do usuário. Para tal, a variável é acrescentada no final do arquivo `.bashrc`, localizado no diretório `/home/<usuário>`. Esse arquivo pode ser visualizado ao digitar no terminal do Linux o comando `ls -a`.

```
# cd /home/<usuário>
# nano .bashrc
(...)
export GASCRP=/usr/local/lib/grads
```

Após a edição, salve e atualize o arquivo:

```
# source .bashrc
```

Obs: 1) Em algumas distribuições Linux pode ser necessário criar um *link* simbólico com o arquivo `libtermcap.so`, localizada dentro do diretório `/usr/lib` (Prestrelo, 2011):

```
# ln -s /usr/lib/libtermcap.so /usr/lib/libtermcap.so.2
```

2) Outros arquivos do GrADS, disponibilizados no sítio do GrADS e/ou criados por seus usuários, também podem ser copiados para o diretório `/usr/local/lib/grads`.

4. CONFIGURAÇÕES INICIAIS DO WRF

As configurações dos domínios e das resoluções horizontais e verticais escolhidas para as simulações do WRF são semelhantes às do MM5 instalado no ICEA. O WRF fornece previsões das condições do tempo a cada 3 horas por um período de 72 horas (3 dias) a partir das 0000UTC e 1200UTC. Os prognósticos são gerados para quatro domínios diferentes, cada qual com uma grade aninhada. Assim, para a mesma região de interesse existem duas grades, entretanto com resoluções distintas (D1 e D2). O domínio denominado de NORTE está centrado na latitude 5°S e longitude de 64°W e apresenta duas grades com resolução horizontal de 36 e 12 km (Fig. 9a e b). O domínio NORDESTE está centrado em 8°S e 38°W e grades com resolução horizontal de 45 e 15 km (Fig. 9c e d). O SUDESTE, centrado em 23,50° S e 46,50° W e grades de 36 e 12 km (Fig. 9e e f). Por fim, o domínio denominado de ALCÂNTARA foca a região do

Centro de Lançamento de Alcântara (CLA) e está centrado na longitude de 02, 40°S e longitude 44, 40°W, apresentando duas grades com resolução horizontal de 18 e 6 km (Fig. 9g e h). A escolha dos domínios está associada ao projeto do Comando da Aeronáutica cujo objetivo é de pesquisa e desenvolvimento da modelagem numérica para fins aeronáuticos e aeroespaciais.

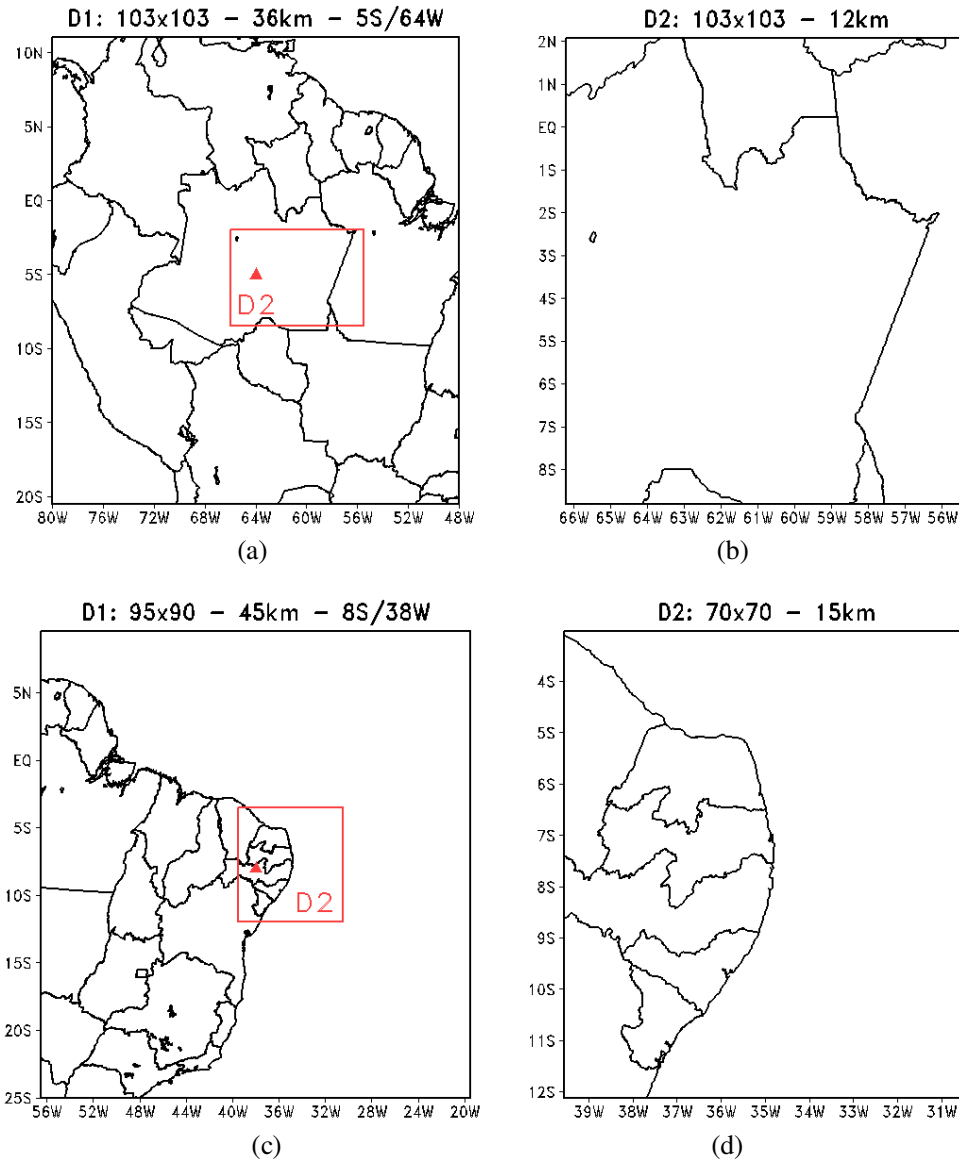


Figura 9 – Domínio das grades NORTE com resolução de 36 km (a) e 12 km (b); NORDESTE com resolução de 45 km (c) e 15 km (d); SUDESTE com resolução de 36 km (e) e 12 km (f); ALCANTARA com resolução de 18 km (g) e 6 km (h). O triângulo vermelho representa o centro da grade principal. (Continua)

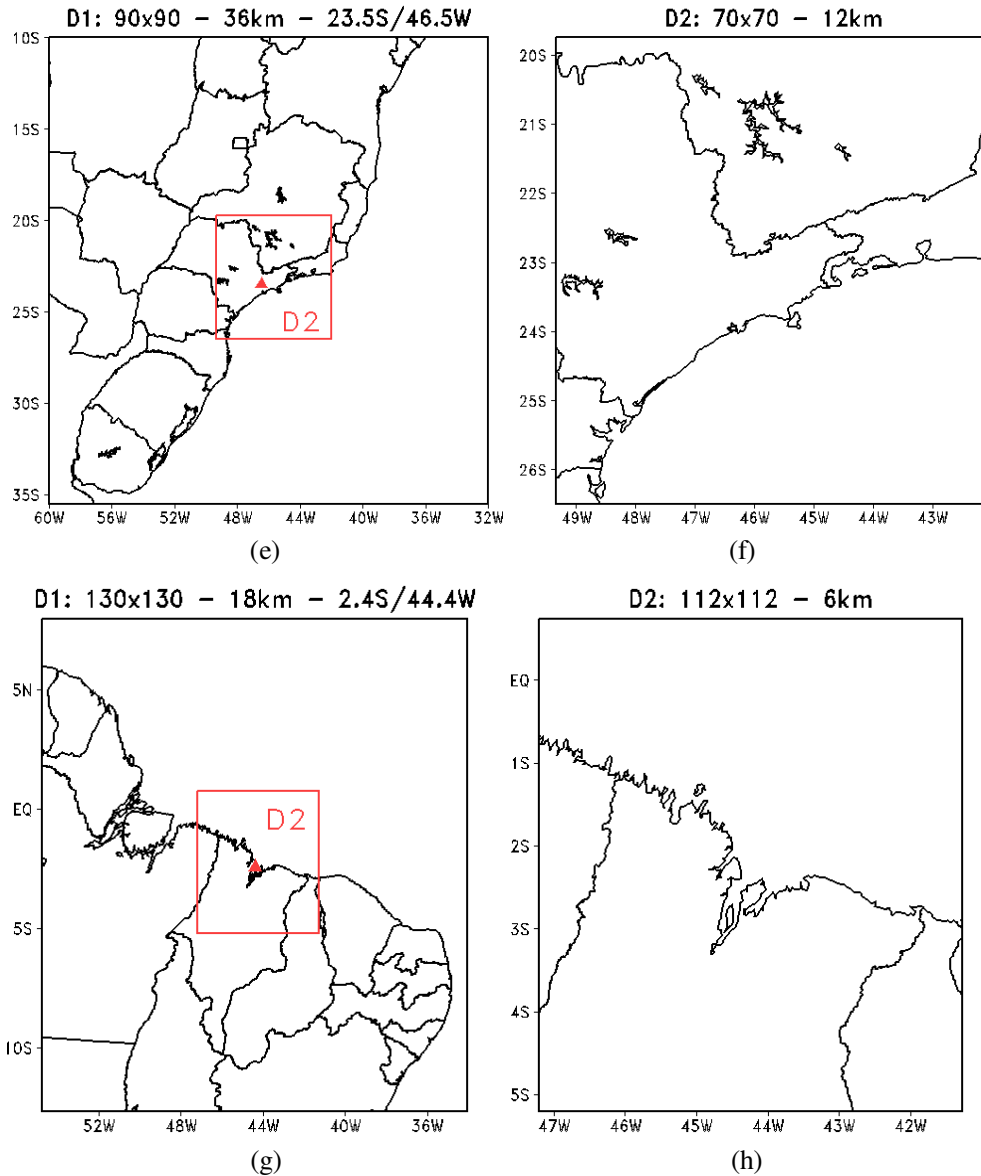


Figura 9 – Domínio das grades NORTE com resolução de 36 km (a) e 12 km (b); NORDESTE com resolução de 45 km (c) e 15 km (d); SUDESTE com resolução de 36 km (e) e 12 km (f); ALCANTARA com resolução de 18 km (g) e 6 km (h). O triângulo vermelho representa o centro da grade principal. (conclusão)

A seguir, é apresentada uma síntese das configurações para cada domínio (Tab. 1).

Tabela 1 – Configurações do modelo WRF.

DOMÍNIO	Resolução horizontal (km)	Nº de pontos zonal x meridional	Extensão do domínio (em graus)
NORTE	36	103 x 103	80-48°W/21°S-12°N
	12	103 x 103	66-55°W/9°S-2°N
NORDESTE	45	95 x 90	56-20°W/24°S-9°N
	15	70 x 70	39-30°W/12°S-3°N
SUDESTE	36	90 x 90	60-32°W/35-10°S
	12	70 x 70	49,5-44°W/26,5-20°S
ALCÂNTARA	18	130 x 130	55-34°W / 12,5°S-8°N
	6	112 x 112	47,5-41,5°W / 5,5°S-0,5°N

A resolução vertical utilizada é de 28 níveis, com pressão no topo da atmosfera de 50 hPa. As parametrizações físicas empregadas no modelo são: WSM3 (HONG, DUDHIA; CHEN, 2004) para microfísica (convecção explícita); RRTM (MLAWER ET AL., 1997) para radiação de onda longa; Dudhia (DUDHIA, 1989) para radiação de onda curta; MM5 similaridade (PAULSON; DYER; HICKS; WEBB, 1970) para camada superfície; Noah-LSM (CHEN; DUBHIA, 2001) para processos de superfície; Yonsei University Scheme (HONG; NOH; DUDHIA, 2006); e Kain-Fritsch (KAIN; FRITSCH, 1990; 1993) convecção rasa e profunda. A escolha das parametrizações físicas segue os valores *default* da instalação do modelo (Tab. 2).

Tabela 2 – Definições adotadas para as simulações iniciais.

parâmetro ou processo físico	valor ou parametrização
microfísica (convecção explícita)	WSM3 (Hong; Dudhia; Chen, 2004)
radiação onda longa	RRTM (Malawer et al., 1997)
radiação de onda curta	Dudhia (Dudhia, 1989)
camada superfície	MM5 similaridade (Paulson et al., 1970)
processos de superfície	Noah-LSM (Chen e Dudhia, 2001)
camada limite planetária	Yonsei University (Hong; Noh; Dudhia, 2006)
convecção rasa e profunda	Kain-Fritsch (Kain; Fritsch, 1990; 1993)

O passo temporal utilizado para D1 (D2) é de 180s (60s). Para o WRF recomenda-se realizar as integrações com passo de tempo equivalente até seis vezes o valor do espaçamento de grade (Δx , em km, ARW V3 *Modeling System User's Guide*, 2011). As condições iniciais e de contorno provêm das análises do modelo global *Global Forecast System* (GFS) do NCEP¹¹ no formato GRIB2, com resolução horizontal de 0,5°; resolução temporal e 3h; e 64 níveis na vertical. O conjunto de dados de tipo de solo e uso do terreno utilizado foi obtido da USGS, dividido em 24 categorias (USGS 24-*category data*).

5. EXECUTANDO O WRF

Para realizar as alterações descritas no tópico anterior, é necessário configurar alguns parâmetros dentro de três arquivos de texto (namelist): *namelist.wps* (pré-processamento), *namelist.input* (processamento) e *namelist.ARWpost* (pós-processamento). Um arquivo *namelist*, codificado em linguagem de programação FORTRAN, contém um conjunto de variáveis para ser lidas durante a execução do programa. O uso do *namelist* permite alterar a configuração do programa sem a necessidade de recompilar o código fonte. Como exemplo, serão mostrados os parâmetros dos *namelist* alterados da grade SUDESTE (Apêndice A). As alterações das outras grades, NORDESTE, NORTE e ALCÂNTARA, podem ser visualizados nos Apêndices B, C, D, respectivamente. Os parâmetros dos arquivos originais, que não forem modificados, serão representados somente por três pontos entre parênteses (...). As alterações dos parâmetros associados ao ano, mês, dia, hora, minuto e segundo do período inicial e final das simulações ocorrerão por meio de um script em SHELL e serão representados pelas palavras YYYY, MM, DD, hh, mm e ss, respectivamente. Todos horários estão em *Coordinated Universal Time* (UTC). O script em SHELL será executado automaticamente por meio do programa CRON. Este programa tem a função de programar a execução de comandos e processos automaticamente nas datas e horários pré-determinados. Neste tópico, são também mostrados os comandos utilizados para executar os subsistemas do WRF (WPS, WRF e WRFpost).

¹¹ <http://nomads.ncep.noaa.gov/pub/data/nccf/com/gfs/prod>

5.1. Execução do WPS

Dentro do diretório WPS, serão editados as variáveis do arquivo namelist.wps. Este arquivo é composto por quatro partes: share, geogrid, ungrid e metgrid. No share, são descritas as variáveis utilizadas por mais que um subsistema do WPS. No geogrid, ungrid e metgrid são especificadas as variáveis para o programa geogrid, ungrid, metgrid. Essas variáveis definem o tamanho e localização de todos os domínios do modelo e dos dados geográficos.

Para armazenar os dados de condições iniciais e de contorno, é criado o diretório GFS2 dentro de /home/WRF/data. Os arquivos GFS têm aproximadamente 55 MB e o seguinte formato: gfs.tHHz.pgrb2fHH, onde a primeira informação horária (HH) indica o início da simulação e o segundo, o período da previsão. Por exemplo, gfs.18Zpgrb2f12, é um arquivo GFS das 12 horas de previsão com hora inicial da simulação às 18Z.

5.1.1 Execução do GEOGRID

Para o GEOGRID, são editadas duas partes do arquivo namelist.wps: share e geogrid. Nesta etapa, é definido o domínio do modelo e interpolado os dados terrestres para as grades do modelo.

```
# cd /home/WRF/WPS
# nano namelist.wps
```

Somente as variáveis mostradas abaixo foram editadas:

```
&share
(...)
max_dom: 2 # número máximo de domínios
(...)
&geogrid
(...)
i_parent_start = 1, 35, # localização inicial da grade aninhada (eixo x)
j_parent_start = 1, 35, # localização inicial da grade aninhada (eixo y)
(...)
```



```

e_we      = 90, 70,    # número de pontos de grade zonais (grade mãe, aninhada)
(...)
e_sn      = 90, 70,    # número de pontos meridionais (grade mãe, aninhada)
geog_data_res = '10m', '5m', # resolução dos dados de terreno (grade mãe, aninhada)
dx        = 36000,    # resolução horizontal no eixo x (grade mãe, em metros)
dy        = 36000,    # resolução horizontal no eixo y (grade mãe, em metros)
map_proj   = 'mercator',
ref_lat    = -23.43,   # latitude do ponto central (grade-mãe)
ref_lon    = -46.47,   # longitude do ponto central (grade-mãe)
(...)
stand_lon  = -46.47,   # longitude padrão paralela ao eixo x da projeção mercator
geog_data_path = '/home/WRF/data/geog', # localização dos dados geográficos
(...)

```

Obs: Não é necessário informar a resolução da grade aninhada ($dx=dy=12000$ m), porque ela segue a razão da grade de 1/3 (*default*), declarada na variável `parent_grid_ratio`.

Antes de executar este subsistema, certifique-se que o arquivo `GEOGRID.TBL` esteja ligado corretamente a sua respectiva versão. Há vários arquivos desse tipo que suportam diferentes núcleos dinâmicos do WRF. `GEOGRID.TBL.ARW` deve ser utilizado para o núcleo ARW.

```

# ls -l geogrid/GEOGRID.TBL
GEOGRID.TBL -> GEOGRID.TBL.ARW

```

Agora, executa-se o `geogrid.exe`. Se a operação for realizada sem falhas, é mostrada na última linha uma mensagem que informará o êxito da operação na última linha do arquivo `log`:

```

# ./geogrid.exe>& geogrid.log
*** Successful completion of program geogrid.exe ***

```

Execute o comando `ls -ltr geo*.nc`. O resultado esperado é: `geo_em_d01.nc` e `geo_em.d02.nc`. Estes são os arquivos da grade criada (um arquivo para cada domínio), após a execução do `geogrid.exe`. Se não forem feitas alterações na grade, o uso do `geogrid.exe` não será mais necessário.

5.1.2 Execução do UNGRIB

Após a criação das grades e interpolado os dados de terreno com o `geogrid.exe`, executa-se o UNGRIB. O UNGRIB extrai os campos meteorológicos do formato GRIB2 necessários para inicializar o modelo. Para examinar se os arquivos GRIB podem ser lidos, utilize o `g2print.exe` (caso os arquivos estejam no formato GRIB 1, utiliza-se o `g1print.exe`):

```
# ./util/g2print.exe /home/WRF/data/GFS2/YYYYMMDDHH/< arquivo GFS >
```

Para o UNGRIB, somente o `share` e `ungrib` necessitam estar editados no `namelist.wps`:

```
&share
(...)
start_date = DATA/HORA, DATA/HORA, # data/hora inicial simulação das grades
        mãe e aninhada (ex.: '2011-01-10_00:00:00', '2011-01-10_00:00:00')
end_date = DATA/HORA, DATA/HORA, # data/hora final simulação das grades mãe e
        inicial da aninhada (ex.: '2011-01-13_00:00:00', '2011-01-10_00:00:00')
interval_seconds = 10800, # frequência dos arquivos de saídas (em s). Modelo
        configurado para gerar arquivos de saídas a cada 3 h.
(...)
&ungrib
(...)
prefix = GFS2, # prefixo para os arquivos intermediários (ex.: GFS2:ANO-MES-
        DIA_HORA). Pode incluir um caminho.
```

O próximo passo é criar os *links* simbólicos para informar a localização dos arquivos GFS e dos arquivos `Vtable`. Os arquivos `Vtable` informam os campos meteorológicos a serem extraídos dos arquivos GRIB2. Alguns `Vtables` estão localizados no diretório `WPS/ungrib/Variable_Tables` (p.ex.: `Vtable.GFS`, `Vtable.SST`, `Vtable.ECMWF`).

```
# ln -s ungrib/Variable_Tables/Vtable.GFS Vtable
# ls -l Vtable
Vtable -> ungrib/Variable_Tables/Vtable.GFS
```

Após os *links* estarem prontos, os seguintes arquivos são visualizados: GRIBFILE.AAA, GRIBFILE.BBB, GRIBFILE.CCC, etc (um para cada arquivo GFS). O *script* link_grib.csh pode ser usado para ligar esses arquivos aos arquivos GRIB.

```
# ./link_grib.csh /home/WRF/data/GFS2/YYYYMMDDhh/gfs*
# ls -l GRIBFILE.*
GRIBFILE.AAA -> /home/WRF/data/GFS2/YYYYMMDDhh/<nome.gfs>
```

Executado o ungrid, se a operação for realizada sem falhas, é mostrada uma mensagem que informará o êxito da operação na última linha do arquivo log.

```
# ./ungrid.exe >& ungrid.log
“*** Successful completion of program ungrid.exe ***”.
```

Serão criados arquivos com o seguinte formato GFS2:YYYY-MM-DD_hh (p.ex.: GFS2:2011-01-11_00). Para examinar se os arquivos intermediários podem ser lidos, utilize o rd_intermediate.exe:

```
#!/util/rd_intermediate.exe GFS2: YYYY-MM-DD_hh
```

5.1.2 Execução do METGRID

O último subsistema do pós-processamento a ser executado é o METGRID. A sua função é interpolar horizontalmente os campos meteorológicos extraídos pelo ungrid.exe para as grades do modelo. Os dados de entrada do METGRID são os arquivos geo_em.dxx.nc (arquivos de saída do GEOGRID); e os arquivos de saída intermediários (provenientes do UNGRIB). Para o METGRID, somente o share e metgrid necessitam estar editados no namelist.wps:

```
&share
(...)
start_date = DATA/HORA, DATA/HORA, # data/hora inicial simulação das grades
        mãe e aninhada (ex.: '2011-01-10_00:00:00', '2011-01-10_00:00:00')
end_date = DATA/HORA, DATA/HORA, # data/hora final simulação das grades mãe e
        inicial da aninhada (ex.: '2011-01-13_00:00:00', '2011-01-10_00:00:00')
(...)
&metgrid
fg_name = 'GFS2',    # prefixo para os arquivos intermediários
```

(...)

```
opt_output_from_metgrid_path = 'home/WRF/WRFV3/test/em_real' # caminho para saídas dos arquivos intermediários
```

Obs.: Sugere-se que a grade aninhada tenha tempos iniciais e finais idênticos ao tempo inicial desejado para a grade aninhada durante a execução do WPS. Isto ocorre porque a grade aninhada obtém as suas condições de contorno laterais da sua grade mãe, e, assim, somente o tempo inicial para uma grade aninhada necessita ser processado pelo WPS, exceto quando for usada análise nudging no WRF.

Antes de executar o METGRID, certifique-se que METGRID.TBL esteja com o *link* correto.

```
# ls -l metgrid/METGRID.TBL
METGRID.TBL -> METGRID.TBL.ARW
```

Para executá-lo, digite o comando:

```
# ./metgrid.exe >& metgrid.log
***** Successful completion of program metgrid.exe *****
```

Os dados de saída do metgrid são:

- met_em.d01.YYYY-MM_DD_hh_mm.nc – um arquivo por tempo; e
- met_em.dxx. YYYY-MM_DD_hh_mm.nc – um arquivo por aninhamento, somente para a hora inicial. Estes arquivos podem ser checados com a ferramenta ncdump (esta ferramenta encontra-se nos arquivos do netcdf):

```
# /usr/local/netcdf-3.6.3/bin/ncdump -h met_em.d01.YYYY-MM_DD_hh_mm.nc
```

5.2 Execução do real e WRF

Este tópico mostrará a execução de dois subsistemas do WRF: o real e o WRF. O trabalho de interpolar verticalmente os campos do WRF é feito dentro do programa real. O WRF é responsável pela execução do modelo propriamente dito. Nesta etapa, o primeiro passo, é editar o arquivo de texto namelist.input que se encontra dentro do diretório /home/WRF/WRFV3/test/em_real.

```
# cd /home/WRF/WRFV3/test/em_real
# nano namelist.input
```

Neste arquivo, são definidos parâmetros como o período da simulação (time_control), a configuração das grades (domains) e as parametrizações físicas (physics).

```
&time_control
run_days   = 0, # dias simulados (Somente grade mãe)
run_hours  = 72, # horas simuladas (Somente grade mãe)
(...)
start_year = YYYY, YYYY, # ano inicial grade mãe, aninhada (ex. 2011)
start_month = MM, MM, # mês inicial grade mãe, aninhada (ex. 01)
start_day   = DD, DD, # dia inicial grade mãe, aninhada (ex. 10)
start_hour  = hh, hh, # hora inicial grade mãe, aninhada (ex. 00)
start_minute = mm, mm, # minutos inicial grade mãe, aninhada (ex. 00)
start_second = ss, ss, # segundos inicial grade mãe, aninhada (ex. 00)
end_year    = YYYY, YYYY, # ano final grade mãe, aninhada (ex. 2011)
end_month   = MM, MM, # mês final grade mãe, aninhada (ex. 01)
end_day     = DD, DD, # dia inicial grade mãe, aninhada (ex. 13)
end_hour    = hh, hh, # hora inicial grade mãe, aninhada (ex. 00)
end_minute  = mm, mm, # minutos inicial grade mãe, aninhada (ex. 00)
end_second  = ss, ss, # segundos inicial grade mãe, aninhada (ex. 00)
interval_seconds = 10800, # frequência dos arquivos de saídas (em s). Modelo
configurado para gerar arquivos de saídas a cada 3 h.
(...)
input_from_file = .true.,true., # Variável lógica, se uma grade aninhada exigir um
arquivo de entrada de dados (p. ex. wrfinput_d02).
(...)
&domains
(...)
max_dom    = 2, # número de domínios (grade mãe e 1 grade aninhada)
(...)
e_we      = 90, 70, # número de pontos na direção x (oeste-leste)
(...)
e_sn      = 90, 70, # número de pontos na direção y (sul-norte)
(...)
```

```

e_vert      = 28, 28,      # número de pontos na direção z (vertical); a dimensão
vertical deve ser a mesma para todas as grades
(...)
dx          = 36000, 12000, # comprimento da grade no eixo x (em m)
dy          = 36000, 12000, #comprimento da grade no eixo y (em m). Nota: dx=dy
(...)
i_parent_start = 1, 35,      # início da grade ponto I (I grade mãe, I grade aninhada)
j_parent_start = 1, 35,      # início da grade ponto J (J grade mãe, J grade aninhada)
(...)

```

Obs.: Foram mantidas no namelist.input original parâmetros importantes, como por exemplo, o espaço temporal (180s) e as parametrizações físicas (Tab. 1). Futuramente, serão realizados testes de sensibilidade para encontrar o conjunto de parametrizações disponíveis no modelo que melhor represente os sistemas atmosféricos atuantes nos domínios simulados.

Com o namelist.input editado, executa-se o arquivo real.exe:

```

# ./real.exe >& real.log
****Successful completion real.exe****

```

Após a execução do real.exe, dois arquivos são criados: wrfinput_d01 e wrfinput_d02. A criação desses arquivos confirma que o modelo está pronto para ser executado. Testes podem ser realizados para verificar o seu conteúdo:

```

# /usr/local/netcdf-3.6.3/bin/ncdump -h wrfinput_d01
# ncdump -v Times wrfinput

```

Por fim, a execução do modelo é dada com o seguinte comando:

```

# ./wrf.exe >& wrf.log
****Successful completion real.exe****

```

Finalizada a execução, é criado dois novos arquivos (um para cada domínio): wrfout_d01_YYY-MM-DD_hh:mm:ss, wrfout_d02_YYY-MM-DD_hh:mm:ss. Estes arquivos são transferidos para o diretório /home/WRF/ARWpost.

```

# mv wrfout* /home/WRF/ ARWpost /

```

5.3 Execução do ARWpost

Executado o modelo, entraremos na etapa do pós-processamento. O ARWpost é usado para converter as saídas do modelos para um formato que possam ser visualizados por ferramentas gráficas (GrADS e Vis5d) . Para executá-lo, primeiro deve-se editar o arquivo namelist.ARWpost:

```
# cd ../ARWpost
# nano namelist.ARWpost
```

No namelist.ARWpost são definidos a data inicial e final da simulação e o intervalo de tempo das saídas dos modelo (datetime); o nome do arquivo WRF que se deseja visualizar, o nome do arquivo de saída para visualizar no GRADS; os campos meteorológicos (io); e os níveis de pressão (interp). As alterações para a grade mãe (D1) são:

```
&datetime
start_date = 'YYYY-MM-DD_hh:mm:ss',#data/hora inicial (ex.:2011-01-10_00:00:00)
end_date = 'YYYY-MM-DD_hh:mm:ss', # data/hora final (ex.:2011-01-13_00:00:00)
interval_seconds = 10800, # frequência dos arquivos de saídas (em s). Modelo
configurado para gerar arquivos de saídas a cada 3 h.
(...)
&io
(...)
input_root_name = './wrfout_d01_YYYY-MM-DD_hh:mm:ss' # diretório de entrada
dos dados (saídas do WRF)
output_root_name = './wrfd01_YYYY-MM-DD_hh:mm:ss' # diretório de saída dos
dados do ARWpost
plot = 'all_list' # Escolha dos campos a ser processados. “all_list”: campos do arquivo
WRF e listados na variável ‘fields’.
fields = 'cape, cin, mcape, mcin, clfr, dbz, max_dbz, geopt, height, lcl, lfc, pressure, rh,
rh2, theta, tc, tk, td, td2, slp, umet, vmet, u10m, v10m, wdir, wspd, wd10, ws10' # lista
os campos meteorológicos escolhidos pelo usuário. Utilizado somente com a opção
“list” na variável ‘plot’.
(...)
```

&interp

```
interp_method = 1,      # 0 - níveis sigma; -1 – código define “adequados” níveis de
                        altura; 1- usuário define níveis de pressão (hPa) ou altura (km)
interp_levels = 1000., 975., 950., 925., 900., 850., 800., 750., 700., 650., 600., 550.,
500., 450., 400., 350., 300., 250., 200., 150., 100., # só usado se interp_method=1.
Indicar níveis para interpolar em hPa (pressão) ou km (altura acima do nível do mar).
Níveis informados de baixo para cima.
```

Obs.: A variável `fields` informa, respectivamente, os seguintes campos diagnósticos: *Convective Available Potential Energy* (CAPE), *Convective Inhibition* (CIN), CAPE máximo, CIN máximo, fração de nuvens baixas/medias e altas; refletividade, refletividade máxima, geopotencial, altura (em Km), nível de condensação por levantamento (NCL), nível de convecção livre (NCL), pressão (em hPa), umidade relativa, umidade relativa 2m, temperatura potencial, temperatura em °C, temperatura em K, temperatura do ponto de orvalho em °C, temperatura do ponto de orvalho 2m, pressão ao nível do mar, ventos rotacionados para coordenadas terrestres, ventos rotacionados para coordenadas terrestres 10m, direção do vento, velocidade do vento, direção do vento 10m, velocidade do vento 10m. Com a opção “all” na variável `plot` é possível visualizar as 111 variáveis do arquivo WRF.

Após as modificações no `namelist.ARWpost`, execute o `ARWpost.exe`:

```
# ./ARWpost.exe >& ARWpostD1.log
```

Altere somente as variáveis `input_root_name` e `output_root_name` para gerar as saídas do modelo para o domínio aninhado (D2):

```
(...)
input_root_name = './wrfout_d02_ YYYY-MM-DD_hh:mm:ss' # diretório de entrada
dos dados
output_root_name = './wrfd02_ YYYY-MM-DD_hh:mm:ss' # diretório de saída dos
dados
(...)
```

E execute novamente o `ARWpost.exe`:

```
# ./ARWpost.exe >& ARWpostD2.log
```


Concluído a execução do pós-processamento, são gerados para cada domínio um arquivo descritor (.ctl) e um de dados (.dat) que poderão ser visualizadas pelo GrADS: (p.ex.: wrfd1_YYYYMMDDhh.ctl ou .dat; e wrfd2_YYYYMMDDhh.ctl ou .dat).

6. CONSIDERAÇÕES FINAIS

A previsão operacional do tempo vem crescendo naturalmente, numa clara afirmação da sua aplicabilidade em diferentes áreas, em particular nas atividades aeronáuticas e aeroespaciais. O ICEA, em parceria com o CNMA e o IAE, decidiu aperfeiçoar o sistema de previsão de tempo com a utilização do modelo WRF. No âmbito das pesquisas de fenômenos atmosféricos e previsão do tempo, o WRF vem sendo amplamente empregado como ferramenta que representa o estado da arte na previsão numérica do tempo.

A instalação do WRF foi realizada com sucesso nos quatros domínios selecionados, destacando-se por (pela): a) oferecer os mais recentes avanços da física, modelagem numérica e assimilação de dados, pois pertence à última geração de modelos numéricos de previsão de tempo; b) aumentar a performance dos recursos computacionais, em virtude de possuir uma arquitetura de software pronta para o processamento paralelo; c) ausência de custos com a compra de licença para software, pois o WRF é disponibilizado gratuitamente e sua codificação é aberta; e d) flexibilidade quanto a sua instalação e configuração. Futuramente, serão realizados testes de sensibilidade para encontrar o conjunto de parametrizações disponíveis no modelo que melhor represente os sistemas atmosféricos atuantes nos domínios simulados. Realizados os testes, o WRF poderá ser disponibilizado na internet e intranet da Rede de Meteorologia do Comando da Aeronáutica (REDEMET), com o intuito de mostrar os produtos meteorológicos gerados pelo modelo.

Espera-se que o presente trabalho tenha contribuído para uma melhor compreensão das características e dos procedimentos necessários para a instalação, compilação e configuração do WRF. Informação detalhada e precisa da utilização desse modelo é necessária para o aperfeiçoamento do sistema de previsão de tempo, tornando-se útil para serem aplicadas nas atividades de interesse do Comando da Aeronáutica, tais como gerenciamento do fluxo de tráfego aéreo e lançamento de foguetes.

7. AGRADECIMENTOS

Ao Grupo de Trabalho (GT) do Programa de Modelagem Numérica do Tempo (PMNT) pela iniciativa do desenvolvimento deste projeto. Ao ICEA, pela disponibilização de sua infraestrutura e dos recursos computacionais. Ao IAE, em especial aos pesquisadores Drº. Gilberto Fernando Fisch e Drº. Marcos Daisuke Oyama pelas valiosas sugestões. Com o intuito de aperfeiçoar este manual, quaisquer sugestões poderão ser encaminhadas para o e-mail: pntar@icea.gov.br.

Citação: IRIART, P. G.; CARVALHO, M. V. C; PEREIRA NETO, A.V. **Manual de instalação, compilação e execução do sistema de modelagem numérica WRF no ICEA.** Subdivisão de Climatologia e Arquivo Meteorológico (PBCA), Instituto de Controle do Espaço Aéreo (ICEA), São José dos Campos, 2011.

REFERÊNCIAS BIBLIOGRÁFICAS

ARW Version 3 Modeling System User's Guide. Weather Research & Forecasting. Mesoscale & Microscale Meteorology Division. National Center for Atmospheric Research. Apr. 2011.

BARROS, S. S. ; OYAMA, M. D. . Sistemas Meteorológicos Associados à Ocorrência de Precipitação no Centro de Lançamento de Alcântara. **Revista Brasileira de Meteorologia**, v. 25, p. 333-344, 2010. ISSN/ISBN: 01027786.

CARVALHO, M. A. V. **Variabilidade da largura e intensidade da Zona de Convergência Intertropical Atlântica:** aspectos observacionais e de modelagem. 107p. Dissertação (Mestrado em Meteorologia) – Instituto Nacional de Pesquisas Espaciais, São José dos Campos, 2011.

CHEN, F.; DUDHIA, J. Coupling an advanced land-surface/ hydrology model with the Penn State/ NCAR MM5 modeling system. Part I: Model description and implementation. **Monthly Weather Review**, v. 129, p. 569-585, 2001.

CORREA, C. S. ; OYAMA, M. D. ; CAMILLO, G. L. ; CALVETTI, L. . Estudo de caso de convecção induzida pela topografia sobre a região sul do Brasil. **Ciência e Natura**, v. 32, p. 83-102, 2010.

DYER, A. J.; HICKS, B. B. Flux-gradient relationships in the constant flux layer, **Quart. J. Roy. Meteor. Soc.**, v. 96, p. 715-721, 1970.

DUDHIA, J. Numerical study of convection observed during the winter monsoon experiment using a mesoscale two-dimensional model. **J. Atmos. Sci**, v. 46, p. 3077-3107, 1989.

HONG, S-Y; DUDHIA, J.; CHEN, S.-H. A revised approach to ice-microphysical processes for the bulk parameterization of clouds and precipitation. **Mon. Wea. Rev.**, 132, 1, 103-120, 2004.

HONG, S.-Y; NOH, Y.; DUDHIA, J. A new vertical diffusion package with an explicit treatment of entrainment processes. **Mon. Wea. Rev.**, v. 134, p. 2318–2341, 2006.

KAIN, J. S.; FRITSCH, J. M. A one-dimensional entraining/detraining plume model and its application in convective parameterization. **J. Atmos. Sci.**, v. 47, n. 23, p. 2748-2802, 1990.

KAIN, J. S.; FRITSCH, J. M. Chapter 16 – Convective parameterization for mesoscale models: The Kain-Fritsch Scheme, in meteorological monographs. **Amer. Meteor. Soc.**, v. 24, n. 46, 1993.

MLAWER, E. J.; TAUBMAN, S. J.; BROWN, P. D.; IACONO, M. J.; CLOUGH, S. A. Radiative transfer for inhomogeneous atmosphere: RRTM, a validated correlated-k model for the longwave. **Journal of Geophysical Research**, v.102, n. D14, p. 16663-16682, 1997.

NOGUEIRA, J. H. A. **Sensibilidade da precipitação simulada na região do Centro de Lançamento de Alcântara a esquemas de convecção explícita: Estudos de Casos**. 2010. 154 p. (INPE-16718-TDI/1656) Dissertação (Mestrado em Meteorologia) - Instituto Nacional de Pesquisas Espaciais, São José dos Campos. 2010.

OYAMA, M.D. **Instalação do modelo de mesoescala MM5 na ACA/IAE/CTA**. São José dos Campos: Divisão de Ciências Atmosféricas, Instituto de Aeronáutica e Espaço, Centro Técnico Aeroespacial, 2003. Publicação interna, PI-002.

OYAMA, M. D. Erros sistemáticos no clima da região tropical da América do Sul simulado pelo modelo regional MM5 em baixa resolução horizontal. **Revista Brasileira de Meteorologia**, v. 21, n. 1, p. 118-133, 2006. ISSN/ISBN: 01027786.

PAULSON, C. A. The mathematical representation of wind speed and temperature profiles in the unstable atmospheric surface layer. **Journal of Applied Meteorology**, v. 9, p. 857-861, 1970.

PEREIRA NETO, A. V. **Avaliação e ajustes da parametrização de Kain-Fritsch sobre a região do Centro de Lançamento de Alcântara**. 2009. 117 p. (INPE-16650-TDI/1613). Dissertação (Mestrado em Meteorologia) - Instituto Nacional de Pesquisas Espaciais, São José dos Campos. 2010.

PRESTELO, C. **Instalação da biblioteca NETCDF e GRADS**. Disponível em: <http://prestelocristiano.webnode.com.br/tutoriais/instalacaodeprogramas/>.

WEBB, E. K. Profile relationships: The log-linear range, and extension to strong stability. **Quart. J. Roy. Meteor. Soc.**, v. 96, p. 67-90, 1970.

APÊNDICE A – NAMELISTS GRADE SUDESTE

➤ **namelist.wps**

```
&share
wrf_core = 'ARW',
max_dom = 2,
start_date = '2011-01-10_00:00:00', '2011-01-10_00:00:00',
end_date = '2011-01-13_00:00:00', '2011-01-10_00:00:00',
interval_seconds = 10800,
io_form_geogrid = 2,
/

&geogrid
parent_id = 1,1,
parent_grid_ratio = 1,3,
i_parent_start = 1,35,
j_parent_start = 1,35,
e_we = 90,70,
e_sn = 90,70,
geog_data_res = '10m','5m',
dx = 36000,
dy = 36000,
map_proj = 'mercator',
ref_lat = -23.43,
ref_lon = -46.47,
truelat1 = 0.0,
truelat2 = 0.0,
stand_lon = -46.47,
geog_data_path = '/home/WRF/data/geog'
/

&ungrib
out_format = 'WPS',
prefix = 'GFS2',
/

&metgrid
fg_name = 'GFS2',
io_form_metgrid = 2,
opt_output_from_metgrid_path = '/home/WRF/WRFV3/test/em_real'
/
```

➤ **namelist.input**

```
&time_control
run_days = 0,
run_hours = 72,
run_minutes = 0,
run_seconds = 0,
start_year = 2011,2011,
start_month = 01,01,
start_day = 10,10,
```

```

start_hour           = 00,00,
start_minute        = 00,00,
start_second        = 00,00,
end_year            = 2011,2011,
end_month           = 01,01,
end_day             = 13,13,
end_hour            = 00,00,
end_minute          = 00,00,
end_second          = 00,00,
interval_seconds    = 10800
input_from_file     = .true.,.true.,
history_interval    = 180, 60,
frames_per_outfile  = 1000,1000,
restart             = .false.,
restart_interval    = 5000,
io_form_history     = 2
io_form_restart     = 2
io_form_input       = 2
io_form_boundary    = 2
debug_level        = 0
/

```

```

&domains
time_step           = 180,
time_step_fract_num = 0,
time_step_fract_den = 1,
max_dom             = 2,
s_we                = 1, 1,
e_we                = 90, 70,
s_sn                = 1, 1,
e_sn                = 90, 70,
s_vert              = 1, 1,
e_vert              = 28, 28,
p_top_requested     = 5000,
num_metgrid_levels  = 27,
num_metgrid_soil_levels = 4,
dx                  = 36000, 12000,
dy                  = 36000, 12000,
grid_id             = 1, 2,
parent_id           = 0, 1,
i_parent_start      = 1, 35,
j_parent_start      = 1, 35,
parent_grid_ratio    = 1, 3,
parent_time_step_ratio = 1, 3,
feedback            = 1,
smooth_option       = 0
/

```

```

&physics
mp_physics          = 3, 3,
ra_lw_physics       = 1, 1,
ra_sw_physics       = 1, 1,
radt                = 30,30,
sf_sfclay_physics  = 1, 1,
sf_surface_physics  = 2, 2,
bl_pbl_physics      = 1, 1,

```

```

bldt = 0, 0,
cu_physics = 1, 1,
cudt = 5, 5,
isfflx = 1,
ifsnow = 0,
icloud = 1,
surface_input_source = 1,
num_soil_layers = 4,
sf_urban_physics = 0, 0,
maxiens = 1,
maxens = 3,
maxens2 = 3,
maxens3 = 16,
ensdim = 144,
/

&fdda
/

&dynamics
w_damping = 0,
diff_opt = 1,
km_opt = 4,
diff_6th_opt = 0, 0,
diff_6th_factor = 0.12, 0.12,
base_temp = 290.
damp_opt = 0,
zdamp = 5000., 5000.,
dampcoef = 0.2, 0.2,
khdif = 0, 0,
kvdif = 0, 0,
non_hydrostatic = .true., .true.,
moist_adv_opt = 1, 1,
scalar_adv_opt = 1, 1,
/

&bdy_control
spec_bdy_width = 5,
spec_zone = 1,
relax_zone = 4,
specified = .true., .true.,
nested = .false., .false.,
/

&grib2
/

&namelist_quilt
nio_tasks_per_group = 0,
nio_groups = 1,
/

```

➤ **namelist. ARWpost**

```
&datetime
start_date = '2011-01-10_00:00:00',
end_date = '2011-01-13_00:00:00',
interval_seconds = 10800,
tacc = 0,
debug_level = 0,
/

&io
io_form_input = 2,
input_root_name = './wrfout_d02_2011-01-10_00:00:00'
output_root_name = './wrfd2_2011011000'
plot = 'all_list'
fields = 'height, geopt, theta, tc, td2, td, rh, rh2, umet, vmet,
slp, pressure, u10m, v10m, wdir, wspd, wd10, ws10, cape, mcin,
lcl, lfc, cape, cin, clfr'
output_type = 'grads'
mercator_defs = .true.
/

&interp
interp_method = 1,
interp_levels = 1000., 975., 950., 925., 900., 850., 800., 750.,
700., 650., 600., 550., 500., 450., 400., 350., 300., 250.,
200., 150., 100.,
/
```

APÊNDICE B – NAMELISTS GRADE NORDESTE

➤ **namelist.wps**

```
&share
wrf_core = 'ARW',
max_dom = 2,
start_date = '2011-05-21_12:00:00','2011-05-21_12:00:00',
end_date = '2011-05-24_12:00:00','2011-05-21_12:00:00',
interval_seconds = 10800,
io_form_geogrid = 2,
/

&geogrid
parent_id = 1,1,
parent_grid_ratio = 1,3,
i_parent_start = 1,44,
j_parent_start = 1,35,
e_we = 95,70,
e_sn = 90,70,
geog_data_res = '10m''5m',
dx = 45000,
dy = 45000,
map_proj = 'mercator',
ref_lat = -8.00,
ref_lon = -38.00
truelat1 = 0.0,
truelat2 = 0.0,
stand_lon = -38.00
geog_data_path = '/home/WRF/data/geog'
/

&ungrib
out_format = 'WPS',
prefix = 'GFS2',
/

&metgrid
fg_name = 'GFS2',
io_form_metgrid = 2,
opt_output_from_metgrid_path = '/home/WRF/WRFV3/test/em_real'
/
```

➤ **namelist.input**

```
&time_control
run_days = 0,
run_hours = 72,
run_minutes = 0,
run_seconds = 0,
start_year = 2011,2011,
start_month = 05,05,
start_day = 21,21,
```



```

start_hour           = 12,12,
start_minute        = 00,00,
start_second        = 00,00,
end_year            = 2011,2011,
end_month           = 05,05,
end_day             = 24,24,
end_hour            = 12,12,
end_minute          = 00,00,
end_second          = 00,00,
interval_seconds    = 10800
input_from_file     = .true.,.true.,
history_interval    = 180, 60,
frames_per_outfile  = 1000,1000,
restart             = .false.,
restart_interval    = 5000,
io_form_history     = 2
io_form_restart     = 2
io_form_input       = 2
io_form_boundary    = 2
debug_level         = 0
/

```

```

&domains
time_step           = 180,
time_step_fract_num = 0,
time_step_fract_den = 1,
max_dom             = 2,
s_we                = 1, 1,
e_we                = 95, 70,
s_sn                = 1, 1,
e_sn                = 90, 70,
s_vert              = 1, 1,
e_vert              = 28, 28,
p_top_requested     = 5000,
num_metgrid_levels  = 27,
num_metgrid_soil_levels = 4,
dx                  = 45000, 15000,
dy                  = 45000, 15000,
grid_id             = 1, 2,
parent_id           = 0, 1,
i_parent_start      = 1, 44,
j_parent_start      = 1, 35,
parent_grid_ratio    = 1, 3,
parent_time_step_ratio = 1, 3,
feedback            = 1,
smooth_option       = 0
/

```

```

&physics
mp_physics          = 3, 3,
ra_lw_physics       = 1, 1,
ra_sw_physics       = 1, 1,
radt                = 30,30,
sf_sfclay_physics  = 1, 1,
sf_surface_physics  = 2, 2,
bl_pbl_physics      = 1, 1,

```

```

bldt = 0, 0,
cu_physics = 1, 1,
cudt = 5, 5,
isfflx = 1,
ifsnow = 0,
icloud = 1,
surface_input_source = 1,
num_soil_layers = 4,
sf_urban_physics = 0, 0,
maxiens = 1,
maxens = 3,
maxens2 = 3,
maxens3 = 16,
ensdim = 144,
/

&fdda
/

&dynamics
w_damping = 0,
diff_opt = 1,
km_opt = 4,
diff_6th_opt = 0, 0,
diff_6th_factor = 0.12, 0.12,
base_temp = 290.
damp_opt = 0,
zdamp = 5000., 5000.,
dampcoef = 0.2, 0.2,
khdif = 0, 0,
kvdif = 0, 0,
non_hydrostatic = .true., .true.,
moist_adv_opt = 1, 1,
scalar_adv_opt = 1, 1,
/

&bdy_control
spec_bdy_width = 5,
spec_zone = 1,
relax_zone = 4,
specified = .true., .true.,
nested = .false., .false.,
/

&grib2
/

&namelist_quilt
nio_tasks_per_group = 0,
nio_groups = 1,
/

```

➤ **namelist. ARWpost**

```
&datetime
start_date = '2011-05-21_12:00:00',
end_date = '2011-05-24_12:00:00',
interval_seconds = 10800,
tacc = 0,
debug_level = 0,
/

&io
io_form_input = 2,
input_root_name = './wrfout_d02_2011-05-21_12:00:00'
output_root_name = './wrfd2_2011052112'
plot = 'all_list'
fields = 'height, geopt, theta, tc, td2, td, rh, rh2, umet, vmet,
slp, pressure, u10m, v10m, wdir, wspd, wd10, ws10, cape, mcin,
lcl, lfc, cape, cin, clfr'
output_type = 'grads'
mercator_defs = .true.
/

&interp
interp_method = 1,
interp_levels = 1000., 975., 950., 925., 900., 850., 800., 750.,
700., 650., 600., 550., 500., 450., 400., 350., 300., 250.,
200., 150., 100.,
/
```

APÊNDICE C – NAMELIST GRADE NORTE

➤ **namelist.wps**

```
&share
wrf_core = 'ARW',
max_dom = 2,
start_date = '2011-05-21_12:00:00','2011-05-21_12:00:00',
end_date = '2011-05-24_12:00:00','2011-05-21_12:00:00',
interval_seconds = 10800,
io_form_geogrid = 2,
/

&geogrid
parent_id = 1,1,
parent_grid_ratio = 1,3,
i_parent_start = 1,45,
j_parent_start = 1,40,
e_we = 103,103,
e_sn = 103,103,
geog_data_res = '10m','5m',
dx = 36000,
dy = 36000,
map_proj = 'mercator',
ref_lat = -5.00,
ref_lon = -64.00,
truelat1 = 0.0,
truelat2 = 0.0,
stand_lon = -64.00,
geog_data_path = '/home/WRF/data/geog'
/

&ungrib
out_format = 'WPS',
prefix = 'GFS2',
/

&metgrid
fg_name = 'GFS2',
io_form_metgrid = 2,
opt_output_from_metgrid_path = '/home/WRF/WRFV3/test/em_real'
/
```

➤ **namelist.input**

```
&time_control
run_days = 0,
run_hours = 72,
run_minutes = 0,
run_seconds = 0,
start_year = 2011,2011,
start_month = 05,05,
```

```

start_day           = 21,21,
start_hour          = 12,12,
start_minute        = 00,00,
start_second        = 00,00,
end_year            = 2011,2011,
end_month           = 05,05,
end_day             = 24,24,
end_hour            = 12,12,
end_minute          = 00,00,
end_second          = 00,00,
interval_seconds    = 10800
input_from_file     = .true.,.true.,
history_interval    = 180, 60,
frames_per_outfile  = 1000,1000,
restart             = .false.,
restart_interval    = 5000,
io_form_history     = 2
io_form_restart     = 2
io_form_input       = 2
io_form_boundary    = 2
debug_level        = 0
/

```

```

&domains
time_step           = 180,
time_step_fract_num = 0,
time_step_fract_den = 1,
max_dom             = 2,
s_we                = 1, 1,
e_we                = 103,103,
s_sn                = 1, 1,
e_sn                = 103,103,
s_vert              = 1, 1,
e_vert              = 28, 28,
p_top_requested     = 5000,
num_metgrid_levels  = 27,
num_metgrid_soil_levels = 4,
dx                  = 36000, 12000,
dy                  = 36000, 12000,
grid_id             = 1, 2,
parent_id           = 0, 1,
i_parent_start      = 1, 45,
j_parent_start      = 1, 40,
parent_grid_ratio    = 1, 3,
parent_time_step_ratio = 1, 3,
feedback            = 1,
smooth_option       = 0
/

```

```

&physics
mp_physics          = 3, 3,
ra_lw_physics       = 1, 1,
ra_sw_physics       = 1, 1,
radt                = 30,30,
sf_sfclay_physics   = 1, 1,
sf_surface_physics  = 2, 2,

```

```

bl_pbl_physics      = 1, 1,
bltd                = 0, 0,
cu_physics          = 1, 1,
cudt               = 5, 5,
isfflx             = 1,
ifsnow             = 0,
icloud             = 1,
surface_input_source = 1,
num_soil_layers    = 4,
sf_urban_physics   = 0, 0,
maxiens            = 1,
maxens             = 3,
maxens2            = 3,
maxens3            = 16,
ensdim             = 144,
/

&fdda
/

&dynamics
w_damping          = 0,
diff_opt           = 1,
km_opt             = 4,
diff_6th_opt       = 0, 0,
diff_6th_factor    = 0.12, 0.12,
base_temp          = 290.,
damp_opt           = 0,
zdamp              = 5000., 5000.,
dampcoef           = 0.2, 0.2,
khdif              = 0, 0,
kvdif              = 0, 0,
non_hydrostatic    = .true., .true.,
moist_adv_opt      = 1, 1,
scalar_adv_opt     = 1, 1,
/

&bdy_control
spec_bdy_width     = 5,
spec_zone          = 1,
relax_zone         = 4,
specified          = .true., .true.,
nested             = .false., .false.,
/

&grib2
/

&namelist_quilt
nio_tasks_per_group = 0,
nio_groups = 1,
/

```

➤ **namelist. ARWpost**

```
&datetime
start_date = '2011-05-21_12:00:00',
end_date = '2011-05-24_12:00:00',
interval_seconds = 10800,
tacc = 0,
debug_level = 0,
/

&io
io_form_input = 2,
input_root_name = './wrfout_d02_2011-05-21_12:00:00'
output_root_name = './wrfd2_2011052112'
plot = 'all_list'
fields = 'height, geopt, theta, tc, td2, td, rh, rh2, umet, vmet,
slp, pressure, u10m, v10m, wdir, wspd, wd10, ws10, cape, mcin,
lcl, lfc, cape, cin, clfr'
output_type = 'grads'
mercator_defs = .true.
/

&interp
interp_method = 1,
interp_levels = 1000., 975., 950., 925., 900., 850., 800., 750.,
700., 650., 600., 550., 500., 450., 400., 350., 300., 250.,
200., 150., 100.,
/
```

APÊNDICE D – NAMELISTS GRADE ALCÂNTARA

➤ **namelist.wps**

```
&share
wrf_core = 'ARW',
max_dom = 2,
start_date = '2011-05-21_00:00:00','2011-05-21_00:00:00',
end_date = '2011-05-24_00:00:00','2011-05-21_00:00:00',
interval_seconds = 10800,
io_form_geogrid = 2,
/

&geogrid
parent_id = 1,1,
parent_grid_ratio = 1,3,
i_parent_start = 1,48,
j_parent_start = 1,48,
e_we = 130,112,
e_sn = 130,112,
geog_data_res = '10m','5m',
dx = 18000,
dy = 18000,
map_proj = 'mercator',
ref_lat = -2.40,
ref_lon = -44.40,
truelat1 = 0.0,
truelat2 = 0.0,
stand_lon = -44.40,
geog_data_path = '/home/WRF/data/geog'
/

&ungrib
out_format = 'WPS',
prefix = 'GFS2',
/

&metgrid
fg_name = 'GFS2',
io_form_metgrid = 2,
opt_output_from_metgrid_path = '/home/WRF/WRFV3/test/em_real'
/
```

➤ **namelist.input**

```
&time_control
run_days = 0,
run_hours = 72,
run_minutes = 0,
run_seconds = 0,
start_year = 2011,2011,
start_month = 05,05,
start_day = 21,21,
```



```

start_hour           = 00,00,
start_minute        = 00,00,
start_second        = 00,00,
end_year            = 2011,2011,
end_month           = 05,05,
end_day             = 24,24,
end_hour            = 00,00,
end_minute          = 00,00,
end_second          = 00,00,
interval_seconds    = 10800
input_from_file     = .true.,.true.,
history_interval    = 180, 60,
frames_per_outfile  = 1000,1000,
restart             = .false.,
restart_interval    = 5000,
io_form_history     = 2
io_form_restart     = 2
io_form_input       = 2
io_form_boundary    = 2
debug_level         = 0
/

```

```

&domains
time_step           = 180,
time_step_fract_num = 0,
time_step_fract_den = 1,
max_dom             = 2,
s_we                = 1, 1,
e_we                = 130, 112,
s_sn                = 1, 1,
e_sn                = 130, 112,
s_vert              = 1, 1,
e_vert              = 28, 28,
p_top_requested     = 5000,
num_metgrid_levels  = 27,
num_metgrid_soil_levels = 4,
dx                  = 18000, 6000,
dy                  = 18000, 6000,
grid_id             = 1, 2,
parent_id           = 0, 1,
i_parent_start      = 1, 48,
j_parent_start      = 1, 48,
parent_grid_ratio    = 1, 3,
parent_time_step_ratio = 1, 3,
feedback            = 1,
smooth_option       = 0
/

```

```

&physics
mp_physics          = 3, 3,
ra_lw_physics       = 1, 1,
ra_sw_physics       = 1, 1,
radt                = 30,30,
sf_sfclay_physics   = 1, 1,
sf_surface_physics  = 2, 2,
bl_pbl_physics      = 1, 1,

```

```

bldt = 0, 0,
cu_physics = 1, 1,
cudt = 5, 5,
isfflx = 1,
ifsnow = 0,
icloud = 1,
surface_input_source = 1,
num_soil_layers = 4,
sf_urban_physics = 0, 0,
maxiens = 1,
maxens = 3,
maxens2 = 3,
maxens3 = 16,
ensdim = 144,
/

&fdda
/

&dynamics
w_damping = 0,
diff_opt = 1,
km_opt = 4,
diff_6th_opt = 0, 0,
diff_6th_factor = 0.12, 0.12,
base_temp = 290.
damp_opt = 0,
zdamp = 5000., 5000.,
dampcoef = 0.2, 0.2,
khdif = 0, 0,
kvdif = 0, 0,
non_hydrostatic = .true., .true.,
moist_adv_opt = 1, 1,
scalar_adv_opt = 1, 1,
/

&bdy_control
spec_bdy_width = 5,
spec_zone = 1,
relax_zone = 4,
specified = .true., .true.,
nested = .false., .false.,
/

&grib2
/

&namelist_quilt
nio_tasks_per_group = 0,
nio_groups = 1,
/

```

➤ **namelist. ARWpost**

```
datetime
start_date = '2011-05-21_00:00:00',
end_date = '2011-05-24_00:00:00',
interval_seconds = 10800,
tacc = 0,
debug_level = 0,
/

&io
io_form_input = 2,
input_root_name = './wrfout_d02_2011-05-21_00:00:00'
output_root_name = './wrfd2_2011052100'
plot = 'all_list'
fields
='height,geopt,theta,tc,td2,td,rh,rh2,umet,vmet,slp,pressure,u10
m,v10m,wdir,wspd,wd10,ws10,cape,mcin,lcl,afc,cape,cin,clfr'
output_type = 'grads'
mercator_defs = .true.
/

&interp
interp_method = 1,
interp_levels =
1000.,975.,950.,925.,900.,850.,800.,750.,700.,650.,600.,550.,500
.,450.,400.,350.,300.,250.,200.,150.,100.,
/
```