

Montar linux box minimo para trabalhar com modelagem 1a aula de lab de modelagem atmosférica

Henrique Barbosa

~~16 abril 2010~~

~~21 abril 2010~~

3 maio 2010

0)Linux

Minha sugestão é que vocês instalem o Ubuntu:

- O ideal é apagar o windows e viver apenas com o linux :-)
- Se não conseguirem, podem separar parte do espaço no HD (mudar as partições) e deixar um espaço para instalar o linux. Há vários tutoriais na internet, fucem!
- Se também for muito complicado, instalem o Ubuntu via Wubi: <http://wubi-installer.org/>. É um programinha para windows que instala o linux dentro do windows. Basicamente ele cria um arquivo com, digamos 15Gb, que vai ser usado como "disco" do ubuntu; e coloca também um opção no menu do windows para decidir se vai inicializar o windows ou o linux.

0.1) Wubi

Na versão 9.10 o Wubi tem um BUG sério. Quando fazemos o update do sistema (atualizamos os pacotes) você para de conseguir entrar no linux! O problema é apenas no programinha que decide se vai entrar no windows ou no linux, mas não tem como entrar no sistema sem corrigir o problema.

Mesmo que não tenha dado problema ainda pra você, se está usando o wubi 9.10, é melhor corrigir logo o defeito. Copie o arquivo: <http://launchpadlibrarian.net/36920146/wubildr> para o c:\ por cima do arquivo que já existe. Feito isso, você pode fazer update sem medo no Wubi 9.10. As instruções completas estão aqui:

http://sourceforge.net/apps/mediawiki/bootinfoscript/index.php?title=Boot_Problems:Wubi_9.10&oldid=214

1)Linux

Para instalar no ubuntu usar o synaptic. Abra o programa, vá em Settings->Repositories. Na aba "Ubuntu Software" marcar todos os itens. Nesta mesma aba, em "Download from", selecionar Brazil e então um dos servidores no Brasil (tem um no IFUSP). Depois do OK, clicar em "Reload" para reler as informações do novo servidor. É possível instalar na mão:

```
$ sudo apt-get install nome_do_pacote
```

Instale os seguintes pacotes:

shells: pdksh, zsh, csh, tcsh, bash

gnu: gcc, g++, gfortran, make, automake, libstdc++5

hdf/nc: h5utils, hdf5-tools, hdf4-tools, nco, netcdf-bin, libnetcdf-dev
graficos: gnuplot, grace (ou xmgr), xfig
outros: emacs, wget, sun-java6-jre
x11: xaw3dg, xaw3dg-dev, libxaw7-dev

Se voce não encontrar o pacote libstdc++5 no repositório, instale a partir do seguinte endereço: <http://packages.ubuntu.com/libstdc++5> a mesma versão do sua distribuição: 32 ou 64bits. Se é uma instalação nova do Ubuntu, veja se não há drivers a serem instalados: System->Administration->Hardware Drivers.

Se o seu sistema é 64bits:

O compilador da intel, que vamos instalar a seguir, foi feito baseado na libstdc++5. Como o Ubuntu (9.10) que estamos instalando já vem com a versão 6, precisamos instalar a versão 5 na mão. Se o seu sistema for 64bits, é preciso instalar as duas versões, 5 e 6, tanto de 32 como de 64! Não me perguntem porque o ifort64 precisa de uma biblioteca de 32bits...

Assim, se seu sistema é 64bits e você **JÁ** instalou a libstdc++6 de 32 **E** 64bits, além da libstdc++5 de 64bits, basta seguir os passos abaixo para resolver o problema:

```
$ cd /usr/lib32
$ ls -l *libstdc*
```

Você deve ver algo assim:

```
lrwxrwxrwx 1 root root      19 Apr 29 2009 libstdc++.so.6 -> libstdc++.so.6.0.10
-rw-r--r-- 1 root root 950424 Mar 16 2009 libstdc++.so.6.0.10
```

Vamos criar um link simbólico (como um atalho do windows) para enganar o compilador:

```
$ ln -s libstdc++.so.6 libstdc++.so.5
```

Veja o que fizemos:

```
$ ls -l *libstdc*

lrwxrwxrwx 1 root root      19 Apr 29 2009 libstdc++.so.6 -> libstdc++.so.6.0.10
-rw-r--r-- 1 root root 950424 Mar 16 2009 libstdc++.so.6.0.10
lrwxrwxrwx 1 root root      19 Apr 29 2009 libstdc++.so.5 -> libstdc++.so.6
```

Apareceu um novo arquivo, chamado libstdc++.so.5 que aponta ("->") para o arquivo que já existia antes, que era justamente a versão 6. Assim, quando o compilador procurar pela versão 5 32bits, esse atalho vai levá-lo a usar o arquivo da versão 6 (sem ele saber!).

2)Compilador Intel: C e F90, por exemplo, para o fortran:

Na pasta CURSO, os arquivos l_cpro**f**_p_11.1.046.tgz e l_cpro**c**_p_11.1.046.tgz são os compiladores da Intel para fortran e C respectivamente. Instale os dois pacotes:

```
$ tar -xzf l_cprof_p_11.1.046.tgz
$ cd l_cprof_p_11.1.046
$ sudo ./install.sh
```

Instale primeiro o fortran (arquivo para o **C** tem um **c** no lugar do **f**).

A instalacao e simples, basta seguir as instruções na tela. Quando for solicitada o numero serial, use o que esta no arquivo "11.1.lic", no diretorio CURSO. Quem não tiver o arquivo, use o sérial: **NDM4-ZVXGRCNN**

Se tiver espaco sobrando, pode instalar a versao completa (Typical Install), caso contrario, escolha a opcao (Custom Install) e selecione apenas os pacotes que voce quer (compilador fortran e debugger). Caso esteja instalando em um sistema de 64bits, voce pode ainda excluir as versoes 32bits dos compiladores. Durante a instalacao eh possivel que apareca uma mensagem reclamando que "operating system type", "glibc" e "binutils" is not supported". Ignore e instale mesmo assim.

Quando for instalar o compilador C, escolha a opcao "Use existing license" para usar a mesma licença do compilador fortran que já está instalada. Caso tenha optado pela "Custom Install", faço o mesmo com o C e escolha apenas o compilador já que o debugger já foi instalado. Como você já instalou o fortran, agora o ./install.sh vai reclamar que o diretório já existe. Pode mandar instalar no mesmo diretório.

Agora os compiladores estão instalados, mas falta ainda informar para o sistema aonde foram instalados. Fazemos isso executando os scripts ifortvars.sh e iccvars.sh que estão em /opt/intel/Compiler/11.1/046/bin/. Se você estiver usando o BASH:

```
$ . /opt/intel/Compiler/11.1/046/bin/ifortvars.sh _versao_
$ . /opt/intel/Compiler/11.1/046/bin/iccvars.sh _versao_
```

onde _versao_ pode ser: 'ia32' (amd ou intel 32bits), 'intel64' (amd ou intel 64bits) ou 'ia64' (itanium). Para você não precisar fazer isso toda vez, é melhor incluir estas linhas no seu ".bashrc"

3)MPICH2: <http://www.mcs.anl.gov/research/projects/mpich2/>

O MPI é uma das bibliotecas que proveem as rotinas necessárias para paralelizar um programa, isto é, para rodar um programa em vários processadores ao mesmo tempo. Para instalar:

```
$ tar -xzf mpich2-1.2.1p1.tar.gz
$ cd mpich2-1.2.1p1
$ ./configure --enable-f77 --enable-f90 --enable-cxx \
  --prefix=/usr/local/intel11.1 CC=icc CXX=icc F77=ifort F90=ifort
$ make
$ make check
$ sudo make install
```

Nota: se você receber uma mensagem de erro como "configure: error: C compiler cannot create executables" você provavelmente esqueceu de instalar o libstdc++5. A versão 11.1 do compilador foi produzida com esta biblioteca, mas o ubuntu 9.10 vem com a versão 6. Veja na página anterior como instalar a versão 5.

4) Bibliotecas

Para instalar o hdf5, é preciso antes instalar o szip e o zlib. As instruções completas de como instala o hdf5 com o compilador da intel estão em:

<http://software.intel.com/en-us/articles/performance-tools-for-software-developers-building-hdf5-with-intel-compilers/>

.1) szip: <http://www.hdfgroup.org/ftp/lib-external/szip/2.1/src/>

```
$ tar -xzvf szip-2.1.tar.gz
$ cd szip-2.1
$ export CC=icc
$ export CXX=icpc
$ export F77=ifort
$ export CFLAGS='-O3 -xP -ip'
$ export CXXFLAGS='-O3 -xP -ip'
$ export FFLAGS='-O3 -xP -ip'
$ ./configure --prefix=/usr/local/intel11.1
$ make
$ make check
$ sudo make install
```

.2) zlib: <http://www.zlib.net/>

```
$ tar -xzvf zlib-1.2.4.tar.gz
$ cd zlib-1.2.4
$ export CC=icc
$ export CFLAGS='-O3 -xP -ip'
$ ./configure --prefix=/usr/local/intel11.1
$ make
$ make check
$ sudo make install
```

.3) hdf5: <http://www.hdfgroup.org/HDF5/>

```
$ tar -xzvf hdf5-1.8.4-patch1.tar.gz
$ cd hdf5-1.8.4-patch1
$ export CC=icc
$ export CXX=icpc
$ export FC=ifort
$ export CFLAGS=''
$ export CXXFLAGS=''
$ export FFLAGS=''
$ ./configure --prefix=/usr/local/intel11.1 \
    --with-zlib=/usr/local/intel11.1 \
    --with-szlib=/usr/local/intel11.1 \
    --enable-fortran --enable-cxx
$ make
$ make check
$ sudo make install
$ cd examples; make; cd..; make check-install
```

.4) netcdf: <http://www.unidata.ucar.edu/downloads/netcdf>

As instruções completas de como instalar o netcdf com os compiladores intel está em: <http://software.intel.com/en-us/articles/performance-tools-for-software-developers-building-netcdf-with-the-intel-compilers/>

```
$ tar -xzvf netcdf-4.1.1.tar.gz
$ cd netcdf-4.1.1
$ export CC=icc
$ export CXX=icpc
$ export CFLAGS='-O3 -xT -ip -no-prec-div -static'
$ export CXXFLAGS='-O3 -xT -ip -no-prec-div -static'

$ export F77=ifort
$ export FC=ifort
$ export F90=ifort
$ export FFLAGS='-O3 -xT -ip -no-prec-div -static'

$ export CPP='icc -E'
$ export CXXCPP='icpc -E'
$ export LD_LIBRARY_PATH=/usr/local/intel11.1/lib:$LD_LIBRARY_PATH

$ ./configure --prefix=/usr/local/intel11.1 \
    --enable-netcdf4 --with-zlib=/usr/local/intel11.1 \
    --with-szlib=/usr/local/intel11.1 --with-hdf5=/usr/local/intel11.1
$ make
$ make check
$ sudo make install
```

5) Programas

.1) Grads: <http://opengrads.org/>

Na pasta CURSO há dois pacotes do grads. O i686 é para sistemas 32bits, enquanto o x86_64 é para 64bits.

```
$ tar -xzvf grads-2.0.a7.oqa.3-bundle-i686-pc-linux-gnu.tar.gz
$ cd grads-2.0.a7.oqa.3
$ sudo mv Contents /usr/local/opengrads-2.0.a7
```

Agora é preciso alterar a variável de ambiente PATH para indicar ao sistema onde o grads foi instalado. Edite o arquivo ~/.profile e inclua o seguinte:

```
export PATH=$PATH:/usr/local/grads-2.0.a7
```

Leia o arquivo na mão com: 'source ~/.profile' e tente executar o grads com: 'grads'. Se ele reclamar de alguma biblioteca que não existe, copie a respectiva biblioteca do subdiretório:

Contents/Linux/Versions/2.0.a7.oqa.3/x86_64/libs

para:

Contents/Linux/Versions/2.0.a7.oqa.3/x86_64/gex

Se você instalou a versão 32bits, então ao invés de x86_64 será i686. Mas lembre-se copie apenas a biblioteca faltante.

.2) CDO: <http://www.mpimet.mpg.de/fileadmin/software/cdo/>

Vamos instalar o CDO (climate data operators) usando a versão do hdf5 e do compilador C do próprio Ubuntu. Abra uma nova aba/janela, assim o novo shell não terá definido nenhuma das variáveis CC, F77, etc... usadas acima.

```
$ tar -xzvf cdo-1.4.1.tar.gz
$ cd cdo-1.4.1
$ ./configure --prefix=/usr/local --with-zlib=/usr --with-szlib=/usr\
    --with-hdf5=/usr
$ make
$ sudo make install
```

.3) ncview: http://meteora.ucsd.edu/~pierce/ncview_home_page.html

```
$ mkdir temp; cd temp
$ tar -xzvf ../ncview-1.93g.tar.gz
$ cd ncview-1.93g
$ ./configure --prefix=/usr/local
$ make
$ sudo make install
```

6) Instalar e rodar o brams-serial