First Time User's Guide
(BRAMS Version 4.0)

Alvaro Luiz Fazenda
Demerval Soares Moreira
Eduardo Hidenori Enari
Jairo Panetta
Luiz Flávio Rodrigues

CPTEC
Version 5 of March 2007

MODIFICATION LOG

Version	Date	Purpose of Modification
1	December 2005	Initial Version for BRAMS 3.2
2	April 2006	First Revision
3	May 2006	Second Revision
4	September 2006	Third Revision
5	March 2007	Initial Version for BRAMS 4.0

INDEX

1.1. Purpose and Intended Audience

This document presents basic information on how to use BRAMS, an open source, limited area weather forecast model. It is intended for first time users, as an introduction to BRAMS User's Guide.

It follows the execution of a script file contained in the download of BRAMS First Time Users version from (http://www.cptec.inpe.br/brams/brams4.0/f_time.shtml), inside BRAMS home page (http://www.cptec.inpe.br/brams). This is the best option to get acquainted with BRAMS, since all execution steps (data acquisition, pre-processing, forecast, post-processing and visualization) are covered. The First Time Users version also installs the entire directory structure, input data files required for a run and output files required for verification.

The first section contains the history of BRAMS. The section after that explains how to download BRAMS First Time Users version and how to run the script file. Document proceeds by visiting each step of the script execution, explaining its meaning, input and output files. Further on, users are acquainted to the structure of the namelist file (RAMSIN) that governs most of the run. Once this is mastered, a second sweep over the script unveils details of the execution. Document ends by indicating how to modify forecast length, date, model resolution and limited area location within RAMSIN.

The BRAMS binary execution file invoked by the script and included in the download runs sequentially on any PC architecture (IA32) Linux System. Even if that is not the desired target BRAMS platform, reading and understanding this basic material is a vital step to understand BRAMS.

1.2. History of BRAMS

BRAMS (Brazilian Regional Atmospheric Modeling System) simulates atmospheric circulations on limited geographical area. It has his roots on RAMS (Regional Atmospheric Modeling System). RAMS is a highly versatile numerical model developed by several groups over the years, including the scientists at Colorado State University and ATMET. RAMS is a multipurpose, numerical prediction model designed to simulate atmospheric circulations spanning from large scale down to large eddy simulations (LES) on the planetary boundary layer.

BRAMS is the outcome of a 2002/2003 Research Project sponsored by FINEP (www.finep.gov.br) aimed to produce a version of RAMS (www.atmet.com) tailored to the tropics, to be used in production mode by Brazilian Regional Weather Centers and to be used in research mode by Brazilian Universities. Although software portability is central to RAMS and BRAMS, the project targeted PC Clusters under Linux. Project partners were:

- ATMET (Atmospheric, Meteorological and Environment Technologies, www.atmet.com),
- IME/USP (Institute of Mathematics and Statistics/University of São Paulo, www.ime.usp.br),

- IAG/USP (Astronomy and Geophysics Institute/University of São Paulo, www.iag.usp.br),
- CPTEC/INPE (Center for Weather Forecasts and Climate Studies/National Institute for Space Research, www.cptec.inpe.br)

The FINEP project generated three versions of BRAMS. Further versions were generated by CPTEC. The previous BRAMS version 3.2 is RAMS Version 5.04 plus:

- Shallow Cumulus and New Deep Convection (mass flux scheme with several closures, based on Grell et al., 2002)
- New 1 km vegetation data derived from IGBP 2.0 + IBGE/INPE dataset LEAF-3 with observed parameters for South American biomes
- Heterogeneous Soil Moisture assimilation procedure
- Operational assimilation cycle and Forecast procedure
- SIB2 surface parameterization
- Binary reproducibility (same result for any number of processors)
- Enhanced Portability and Software Quality
- Improved serial and parallel performance

In 2007, a new version of BRAMS was released: BRAMS Version 4.0. These version is the result of the improvements (computational and meteorological) over the BRAMS Version 3.2 plus some corrections based on RAMS Version 6.0. In this way, BRAMS version 4.0 is BRAMS version 3.2 plus:

- Enhanced Portability and Software Quality (new procedure to read RAMSIN, corrections needed by others compilators, new scheme to build executable code, etc).
- Improved serial and parallel performance (best vectorization in same codes, improvements on performance for advection scheme and improvements in master slave comunications).
- Inclusion of TEB (Town Energy Budget) scheme.
- Inclusion of CATT (Coupled Aerosol and Tracer Transport) scheme.
- Corrections in Shaved ETA scheme based on RAMS 6.x.
- Corrections in LEAF scheme based on RAMS 6.x.

BRAMS is maintained and supported by the BRAMS team at CPTEC. Maintenance and support should be obtained by mailing a message to brams@cptec.inpe.br. There are a discussion list: brams_list@cptec.inpe.br. To subscribe this discussion list, visit http://www.cptec.inpe.br/maillist.shtml for more informations. The source code is distributed under the GNU General Public License. For further information, visit BRAMS site at www.cptec.inpe.br/brams.

1.3. Download and run BRAMS First Time Users

BRAMS site contains two tarballs for software download: First Time Users and Expert Users.

The First Time Users download file contains BRAMS as an executable binary file that runs sequentially on PC architecture (x86) systems under Linux. It contains all input data required for a sample run, as well as expected output, so that first time users can check correctness of the installation.

The Expert Users download file contains only the source code and instructions for compilation; it is intended for execution on parallel machines or on systems other than Linux. Since the Expert Users download file does not contain input data, we strongly recommend download and installation of the First Time User tarball even if the target system is not Linux.

Grads (Grid Analysis and Display System) should be installed to visualize BRAMS output. The presence of Grads is not mandatory for BRAMS execution but its absence prevents verification of the installation. Grads can be downloaded from its home page:

```
http://grads.iges.org/grads/grads.html
```

To begin the BRAMS installation, download First Time Users file from:

```
http://www.cptec.inpe.br/brams/run/brams4.0-serial.tar.gz
```

It should be downloaded to a directory of your choice that will serve as BRAMS home directory, such as

```
mkdir /home/user/run-brams40
```

Being at BRAMS home directory, expand the tarball:

```
tar -xzvf brams3.2-serial.tar.gz
```

A directory structure is created with root at the BRAMS home directory. To certify the installation, run BRAMS for the first time:

```
./runBRAMS.sh
```

The `runBRAMS.sh` file is a script that executes the following steps:

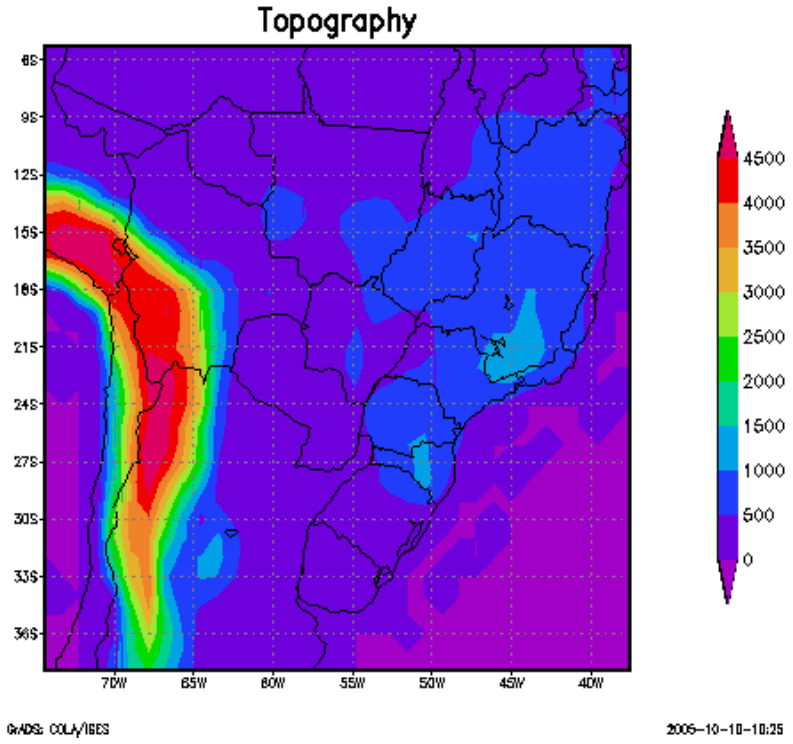
- Convert CPTEC global files into BRAMS input data file format;
- Build surface file for the desired area;
- Build initial and boundary conditions for the desired area;
- Run the sequential forecasting;
- Post process and picture generation

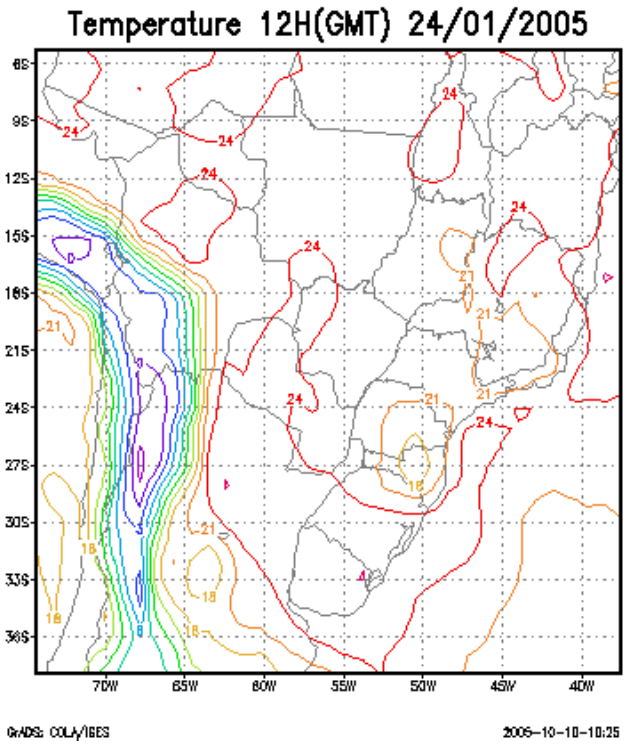
For each of these steps, the script outputs comments. If the run proceeds successfully, script execution ends with the following message:

```
"BRAMS execution ends successfully"
```

If any error occurs during script execution, a warning will be printed and execution halts. In that case, support is available by sending an e-mail to brams@cptec.inpe.br, with as much information as possible.

Successful sample run generates gif files topo.gif and temp.gif at directory RAMSPOST50 just like this:




Temperature 12H(GMT) 24/01/2005
0S
3S
6S
9S
12S
15S
18S
21S
24S
27S
30S
33S
36S
70W
65W
60W
55W
50W
45W
40W
GrADS: COLA/IGES
2005-10-10-10:25

1.4. Understanding BRAMS First Time User Run

Let's investigate how BRAMS works, exploring each step of the script *runBRAMS.sh*, as well as its input and output data. Our first sweep over the script just explains the basics. A second sweep, at section shows details.

Decompression of the tarball *brams4.0-serial.tar.gz* produces a directory hierarchy containing all files required by BRAMS. All output files will also be written into the same directory hierarchy.

Initially, BRAMS home directory has the following files:

```
A/                  dprep/      micro/          RAMSIN-vfile    sst/            veg_usgs/

Brams4.0-serial.x   H/          RAMSIN-initial  RAMSPOST55/     topo/

data/               ivar/       RAMSIN-sfc      runBRAMS.sh*    topo10km/
```

File *brams4.0-serial.x* is BRAMS' executable. Files *RAMSIN-initial, RAMSIN-sfc* and *RAMSIN-vfile* are input data governing BRAMS execution.

Directory *RAMSPOST55* contains post-processing executable, input data files and visualization scripts.

Directories *A* and *H* are initially empty, since they are reserved for BRAMS output forecast (*A* is for model forecasts, known as *analysis*, and *H* for model restart files, known as *history*).

Directories *ivar* and *data* are also initially empty, since they are reserved for BRAMS intermediate output. Remaining directories contain input data for the various stages of BRAMS execution:

- Directory *sst* contains sea surface temperature files.
- Directories *topo* and *topo10km* contain topography files.
- Directory *veg_usgs* contains vegetation files
- Directory *micro* contains a special input microphysics file
- Directory *dprep* contains CPTEC global forecast files.

We proceed by visiting each step of the script *runBRAMS.sh*.

1.4.1 Converting CPTEC global files

First step is to convert files containing the state of the atmosphere over time as forecasted by CPTEC into a file format that BRAMS understands. CPTEC produces files

on *grib* format containing the state of the atmosphere on a Gaussian grid that encompasses South America. BRAMS does not understand this file format – it expects an ASCII file on a regular grid that contains the state of the atmosphere. File conversion is performed by *gribT126_to_dp.x*, the first step of *runBRAMS.sh*. There is a new version of this application in http://www.cptec.inpe.br/brams/utilities_grib2dp.shtml. The new version allows to use CPTEC analysis files that encompasses others regions. Visit the page for more informations.

Input and output data files reside at directory *dprep*. CPTEC global files (input to *gribT126_to_dp.x*) are named *GAMRAMSXXXX.grib*. BRAMS input files (output of *gribT126_to_dp.x*) are named *dpYYYY*. Both files span the same domain.

1.4.2 Build surface files for the desired limited area

Second step is to generate topography, sea surface temperature and vegetation cover files for the desired limited area. The script accomplishes this step by executing BRAMS, driven by input file RAMSIN_sfc.

Input data files are those contained at directories *sst, topo10km, topo* and *veg_usgs.*

Resulting files are stored at directory *data*. These are files *sst-brams-W-XXXX*, *ndvi-brams-N-YYYY, sfc-brams-X-ZZZZ* and *toph-brams-S-WWWW*.

1.4.3 Build initial and boundary conditions for the desired limited area

Third step is to generate the initial state of the atmosphere and boundary conditions over the forecast time for the desired limited area. The script accomplishes this step by executing BRAMS, driven by input file RAMSIN_vfile.

Input data files are script's first step output – files *dpYYYY* at directory *dprep.* BRAMS reads these files, cuts their contents to the desired limited area and outputs files *iv-brams-V-ZZZZ,* at directory *ivar.*

1.4.4 Run forecast

Fourth step is to run the forecast for the desired limited area and time period. The script accomplishes this step by executing BRAMS, driven by input file RAMSIN_initial.

Input data files are the output of the last two steps – topography, sea surface temperature, vegetation cover, initial and boundary condition files, all resident on previous steps output directories.

There are two kinds of output data files: *history* and *analysis*.

History files are stored at directory *H*, containing all required information for BRAMS re-start (*checkpoint* files). These are named *hist-H-AAAA.*

Analysis files are stored at directory *A*, containing the state of the atmosphere at desired limited area and time, as forecasted by BRAMS. These are named *anal-A-ZZZZ.*

1.4.5 Post processing and picture generation

Analysis files are not suited for human interpretation. Post processing phase selects desired fields from the Analysis files and produces files suited for visualization. Current script produces files for the Grads visualization tool.

Post processing is performed by the *ramspost55.x* executable at directory *RAMSPOST55*, invoked by the shell command *run.sh*, at the end of the *runBRAMS* script file.

Input files are the Analysis files.

Output files are the *result_XXX* files at *RAMSPOST55* directory.

Picture generation is performed by Grads, taking as input *result_XXX* files and producing as output *topo.gif* and *temp.gif* files, all at *RAMSPOST55* directory. Grads is invoked by script *run.sh*.

1.5. Governing the run: the RAMSIN file

Before proceeding into a deeper understanding of BRAMS execution, it is central to understand how to govern the run.

RAMSIN is the input file that specifies an execution of BRAMS. Observe that *runBRAMS.sh* executes BRAMS three times, each ingesting a specific RAMSIN: RAMSIN_sfc (to build surface files), RAMSIN_vfile (to build initial and boundary conditions) and RAMSIN_initial (to run the forecast).

These three files differ only by the execution mode (variable *RUNTYPE* at namelist MODEL_GRIDS). Another way to achieve the same result is to have a single RAMSIN file, modifying the *RUNTYPE* input variable according to the desired run.

RAMSIN is composed by a set of Fortran 90 namelist data that govern the execution of BRAMS in two modes: as an atmospheric model (BRAMS) and as a data acquisition program (BRAMS ISAN, standing for ISentropic ANalysis). There are a few namelists for each program.

For the atmospheric model component of BRAMS, the namelists are:

- $MODEL_GRIDS
- $MODEL_FILE_INFO
- $MODEL_OPTIONS
- $MODEL_SOUND
- $MODEL_PRINT

In BRAMS version 4.0, two new namelist sections were added to RAMSIM namelist file. These sections are related with the new emission models coupled to the atmospheric model and the namelist are:

- $CATT_INFO
- $TEB_INFO

For ISAN, the namelists are:

- $ISAN_CONTROL
- $ISAN_ISENTROPIC

Each RAMSIN namelist begins with one of the above identifiers and ends with the identifier

$END.

Namelist $MODEL_GRIDS specify the grids and the type of the run. Run type is defined by the value of variable *RUNTYPE* (a Fortran character string) which can take one out of four values: MAKESFC (to build surface files), MAKEVFILE (to build boundary conditions), INITIAL (to run the forecast from initial data) or HISTORY (to resume a previous forecast run, from a history file). Please, observe how this variable is set on the three RAMSIN files corresponding to the three runs of BRAMS invoked by *runBRAMS.sh.*

Forecast time length is specified by variable *TIMMAX* (a Fortran integer). Time unit for the value of *TIMMAX* is set by *TIMEUNIT* (a Fortran character string that takes as value one of "h", "m" or "s"). Forecast initial time is set by variables *IMONTH1*, *IDATE1*, *IYEAR1* and *ITIME1*, all Fortran integers, corresponding to the month, date, year and day time, in hours (24 hours clock), of the data contained on initial files.

Horizontal model grid is specified by variables *NNXP*, *NNYP*, *DELTAX*, *DELTAY*, *CENLAT* and *CENLON*. Observe that each of these variables in any of the three RAMSIN contain three values. That is because BRAMS allows grid nesting, meaning that three grids with increasing resolution but with decreasing geographical area are specified. But only the external grid is used – the two internal grids are not used. That is due to the value of the *NGRIDS* variable (set to 1).

Namelist $MODEL_FILE_INFO specify all input and output files for some of the execution modes of BRAMS and how to interpret them. Take, as an example, variables *HFILOUT* and *AFILOUT* (both Fortran character strings) that specify the prefix of history and analysis files. Variables *FRQHIS* and *FRQANL* (both Fortran real values) specify the frequency of writing history and analysis files.

Namelist $MODEL_OPTIONS select methods for physics, surface and dynamics during forecast. See, for example, variable *LEVEL* (a Fortran integer) that defines the microphysics level of the run, from a very simple microphysics (value 1) to a complex and costly microphysics (value 3).

Namelist $MODEL_SOUND is a short namelist to deal with sounding specification.

Namelist $MODEL_PRINT specifies which fields should be printed on BRAMS standard output and their frequency.

Namelist $CATT_INFO controls the running of the Coupled Aerosol and Tracer Transport (CATT) scheme. It allows activate or not the CATT emission model, and specifies the variables needed in this special case of simulation.

Namelist $TEB_INFO controls the running of the Town Energy Budget (TEB) scheme. It activate or not the TEB emission model, and specifies the variables needed in this special case of simulation.

Note that in this version 4.0 of BRAMS, is not possible to run booth emission model (CATT and TEB) at the same time. This new feature is planed to be available only in future versions.

Namelists $ISAN_CONTROL and $ISAN_ISENTROPIC controls data assimilation, including the insertion of local observations.

These namelists will be extensively referred throughout this document. The Configuration section of this document shows how to modify RAMSIN to fit a specific area of South America and a specific time. For in-depth information, see MODEL INPUT NAMELIST PARAMETERS by Robert L. Walko and Craig J. Tremback (2002 at http://www.atmet.com/html/docs/documentation.shtml).

1.6. A deeper understanding

Our second sweep over the five steps of the script will reveal how each step works and how to drive each run.

1.6.1 Converting CPTEC Global Files

Any limited area forecast model requires knowledge of the state of the atmosphere at the beginning of the computation (for grid initialization) and from time to time (for boundary conditions and, in the case of BRAMS, to nudge the external state of the atmosphere into the internal state).

CPTEC produces daily forecasts of the state of the atmosphere for the entire Earth and for the next 15 days. These are the output of CPTEC's global model, currently running in production mode at the T126L28 configuration (102 km horizontal resolution at the Equator, 28 vertical levels). CPTEC performs a specific post-processing of the global model output for the BRAMS community. Post-processed files are grib formatted and named *GAMRAMS<begin date><current date>P.fct.T126L28.grb*, where *<date>* has the format YYYYMMDDHH (year YYYY, month MM, day DD and hour HH). Each GAMRAMS file spans a region of the Earth covering longitude 120º west to 80º east and latitude 89 º south to 40 º north. Latitudes belong to a Gaussian grid. There are 13 vertical levels (sigma coordinate). First Time Users download provides *GAMRAMS* files at the *dprep* directory from 12:00 of January 24th, 2005 to 18:00 of January 27th, 2005.

BRAMS requires the state of the atmosphere in ASCII format over a regular grid. These files are named *dpYYYY-MM-DD-HHHH*. Converting *GAMRAMS* files to *dp* is the responsibility of the executable *gribT126_to_dp.x*. Residing at the same *dprep* directory, this executable converts every *GAMRAMS* file at this directory into an equivalent *dp* file, as specified by the *PREP_IN* namelist. Run standard output is stored at file *grib_to_dp.out*, at the same directory.

Observe that this first step of the *runBRAMS.sh* script does not use any information from the *RAMSIN* file. Consequently, it does not know the area to be forecasted. It is the user's responsibility to guarantee that the forecasted area lies within the area covered by the *GAMRAMS* and the *dp* files.

1.6.2 Build surface files for the desired limited area

After converting CPTEC global model output files, the second step of *runBRAMS.sh* is to execute BRAMS with RUNTYPE set to MAKESFC (in RAMSIN). In this run type, global data files with sea surface temperature, soil type and topography will

be converted to files that just cover the limited area (and grid) defined by the namelist $MODEL_GRIDS (in RAMSIN).

File *sfc.out* (at BRAMS home directory) stores the standard output of this run.

SST, soil type and topography files are required for each model forecast run. They are usually output of a separate model run (RUNTYPE set to MAKESFC), as in this step of the *runBRAMS.sh* script. If RUNTYPE is instead set to MAKEVFILE or INITIAL, and if they already exist from a prior run and are consistent with model grid size and location (as in this case), they are directly used.

Files produced by this run are stored at directory *data*:

```
-rw-r--r--     1 user users   9907 2005-12-02 13:35 ndvi-brams-N-2005-01-24-000000-g1.vfm
-rw-r--r--     1 user users 106570 2005-12-02 13:35 sfc-brams-S-g1.vfm
-rw-r--r--     1 user users   5022 2005-12-02 13:35 sst-brams-W-0000-01-16-120000-g1.vfm
-rw-r--r--     1 user users   5022 2005-12-02 13:35 sst-brams-W-0000-02-15-000000-g1.vfm
-rw-r--r--     1 user users   5022 2005-12-02 13:35 sst-brams-W-0000-03-16-120000-g1.vfm
-rw-r--r--     1 user users   5022 2005-12-02 13:35 sst-brams-W-0000-04-16-000000-g1.vfm
-rw-r--r--     1 user users   5022 2005-12-02 13:35 sst-brams-W-0000-05-16-120000-g1.vfm
-rw-r--r--     1 user users   5022 2005-12-02 13:35 sst-brams-W-0000-06-16-000000-g1.vfm
-rw-r--r--     1 user users   5022 2005-12-02 13:35 sst-brams-W-0000-07-16-120000-g1.vfm
-rw-r--r--     1 user users   5022 2005-12-02 13:35 sst-brams-W-0000-08-16-120000-g1.vfm
-rw-r--r--     1 user users   5022 2005-12-02 13:35 sst-brams-W-0000-09-16-000000-g1.vfm
-rw-r--r--     1 user users   5022 2005-12-02 13:35 sst-brams-W-0000-10-16-120000-g1.vfm
-rw-r--r--     1 user users   5022 2005-12-02 13:35 sst-brams-W-0000-11-16-000000-g1.vfm
-rw-r--r--     1 user users   5022 2005-12-02 13:35 sst-brams-W-0000-12-16-120000-g1.vfm
-rw-r--r--     1 user users   9922 2005-12-02 13:35 toph-brams-S-g1.vfm
```

Sea Surface Temperature files:

Sea surface temperature (SST) input and output file path and prefix are defined by two variables: *SSTFPFX* and *ISSTFN*, at namelist $MODEL_FILE_INFO (in RAMSIN):

```
SSTFPFX = './data/sst-brams',
ISSTFN = './sst/S',
         './sst/S',
         './sst/S',
```

Input SST filename path and prefix are defined by the *ISSTFN* namelist variable (a Fortran character string). It has three values in sequence, since this RAMSIN is prepared for a nesting of three grids (a value for each grid). These files contain sea surface temperature data for each model grid, and usually for multiple times. BRAMS expects a latitude-longitude grid with climatology values for each month of the year or for each week.

In BRAMS first time users download, directory *sst* has the following files:

```
-rw-r--r-- 1 user users 66420 2002-06-19 10:46 SAPR90S000E
-rw-r--r-- 1 user users 66420 2002-06-19 10:46 SAPR90S180W
-rw-r--r-- 1 user users 66420 2002-06-19 10:46 SAUG90S000E
-rw-r--r-- 1 user users 66420 2002-06-19 10:46 SAUG90S180W
-rw-r--r-- 1 user users 66420 2002-06-19 10:46 SDEC90S000E
-rw-r--r-- 1 user users 66420 2002-06-19 10:46 SDEC90S180W
-rw-r--r-- 1 user users 66420 2002-06-19 10:46 SFEB90S000E
-rw-r--r-- 1 user users 66420 2002-06-19 10:46 SFEB90S180W
-rw-r--r-- 1 user users   433 2004-05-13 13:57 SHEADER
-rw-r--r-- 1 user users 66420 2002-06-19 10:46 SJAN90S000E
-rw-r--r-- 1 user users 66420 2002-06-19 10:46 SJAN90S180W
```

```
-rw-r--r-- 1 user users 66420 2002-06-19 10:46 SJUL90S000E
-rw-r--r-- 1 user users 66420 2002-06-19 10:46 SJUL90S180W
-rw-r--r-- 1 user users 66420 2002-06-19 10:46 SJUN90S000E
-rw-r--r-- 1 user users 66420 2002-06-19 10:46 SJUN90S180W
-rw-r--r-- 1 user users 66420 2002-06-19 10:46 SMAR90S000E
-rw-r--r-- 1 user users 66420 2002-06-19 10:46 SMAR90S180W
-rw-r--r-- 1 user users 66420 2002-06-19 10:46 SMAY90S000E
-rw-r--r-- 1 user users 66420 2002-06-19 10:46 SMAY90S180W
-rw-r--r-- 1 user users 66420 2002-06-19 10:46 SNOV90S000E
-rw-r--r-- 1 user users 66420 2002-06-19 10:46 SNOV90S180W
-rw-r--r-- 1 user users 66420 2002-06-19 10:46 SOCT90S000E
-rw-r--r-- 1 user users 66420 2002-06-19 10:46 SOCT90S180W
-rw-r--r-- 1 user users 66420 2002-06-19 10:46 SSEP90S000E
-rw-r--r-- 1 user users 66420 2002-06-19 10:46 SSEP90S180W
```

Output SST filename path and prefix are defined by the *SSTFPFX* namelist variable (a Fortran character string). Observe that output SST files (at directory *data*) have the prefix sst-brams, as defined by the *SSTFPFX* variable.

Topography and Vegetation Cover files:

Input surface files path and prefix are defined by variables *ITOPTFN*, *IFEGTFN* and *ISOILFN* (all Fortran character strings) at RAMSIN namelist $MODEL_FILE_INFO. The named files contain, respectively, topography, vegetation type and soil textural class:

```
ITOPTFN  = './topo10km/H',
IVEGTFN  = './veg_usgs/VEGET_',
ISOILFN  = ' ',
```

Output surface files path and prefix are defined by the *SFCFILES* variable in RAMSIN namelist $MODEL_FILE_INFO:

```
SFCFILES = './data/sfc-brams',
```

Output surface files contain topography, soil textural class, vegetation type, and subgrid distribution of soil textural class, vegetation type and water surface areas for each model grid. Observe that output files (at directory *data*) have the prefix sfc-brams, as defined by the *SFCFILES* variable.

1.6.3 Build initial and boundary conditions for the desired limited area

The third step of *runBRAMS.sh* is to execute BRAMS with *RUNTYPE* set to MAKEVFILE. This run produces files with initial and boundary conditions for each grid over the integration time. This is performed by cutting the *dp* files (shell's first step output) to the desired geographical areas, grids and time interval, as defined by the namelist $MODEL_GRIDS (in RAMSIN). Output files are known as VARFILES (that's why *RUNTYPE* is set to MAKEVFILE).

File *vfile.out* (at BRAMS home directory) stores the standard output of this run. Directory *ivar* stores files output by this run:

```
-rw-r--r-- 1 user users 597320 2005-11-22 14:28 iv-brams-V-2005-01-24-120000-g1.vfm
-rw-r--r-- 1 user users     13 2005-11-22 14:28 iv-brams-V-2005-01-24-120000.tag
-rw-r--r-- 1 user users 597320 2005-11-22 14:28 iv-brams-V-2005-01-24-180000-g1.vfm
-rw-r--r-- 1 user users     13 2005-11-22 14:28 iv-brams-V-2005-01-24-180000.tag
-rw-r--r-- 1 user users 597320 2005-11-22 14:28 iv-brams-V-2005-01-25-000000-g1.vfm
```

```
-rw-r--r-- 1 user users     13 2005-11-22 14:28 iv-brams-V-2005-01-25-000000.tag
-rw-r--r-- 1 user users 597320 2005-11-22 14:28 iv-brams-V-2005-01-25-060000-g1.vfm
-rw-r--r-- 1 user users     13 2005-11-22 14:28 iv-brams-V-2005-01-25-060000.tag
-rw-r--r-- 1 user users 597320 2005-11-22 14:28 iv-brams-V-2005-01-25-120000-g1.vfm
-rw-r--r-- 1 user users     13 2005-11-22 14:28 iv-brams-V-2005-01-25-120000.tag
```

The aim of the MAKEVFILE run type is to create the initialization and boundary conditions to BRAMS simulation (VARFILES) from the *dp* files generated by the first step (in the *dprep* directory).

Input data files path and prefix are specified by the *IAPR* variable (a Fortran character string) at namelist $ISAN_CONTROL:

```
IAPR    = './dprep/dp',
```

Output data file path and prefix are specified by variable *VARFPFX* (a Fortran character string) at namelist $ISAN_CONTROL:

```
VARFPFX    = './ivar/iv-brams',
```

Observe that the production of *dp* files on the first step of the script is disconnected from the value of the *IAPR* variable, the time of integration and the frequency of boundary conditions. It is the user's responsibility to guarantee that GAMRAMS files cover the desired area, have the desired frequency and that file paths and prefix of the first step output are coherent with the third step input.

Each VARFILE contains the horizontal wind, pressure, potential temperature, and vapor mixing ratio fields interpolated to the model grid(s) and ready for initialization and/or time-dependent data assimilation. The file name begins with the given prefix. To this prefix it is automatically appended the year, month, date and time.

1.6.4 Run forecast

After obtained all input data files (with runs MAKESFC and MAKEVFILE), BRAMS can do the forecast by setting *RUNTYPE* to INITIAL. This setting means that a simulation will begin from time zero and will run up to the time specified by variable *TIMEMAX*, that specify time in *TIMEUNITS* units.

This is the forth step of the script. Standard output of the run is stored at file *initial.out* on BRAMS home directory.

This run assumes that all atmospheric and soil prognostic variables are initialized from a varfile prepared in earlier runs. Varfiles will be obtained from path and prefix specified by the value of the *VARFPFX* variable of the $MODEL_FILE_INFO namelist (as explained in the previous step). Varfiles will be used only if model runs with variable *RUNTYPE* is set to INITIAL or HISTORY and variable *INITIAL* (at $MODEL_FILE_INFO namelist) is set to 2:

```
RUNTYPE = 'INITIAL'

.....
! Variable initialization input
INITIAL = 2, ! Initial fields - 1=horiz.homogeneous,
             ! 2=init from varfile
             ! 3=init from HFILIN
```

There are other ways to initialize the forecast. Variables can be initialized with homogeneous values (if *INITIAL* is set to 1) or interpolated from a previous history file (if *RUNTYPE* is set to HISTORY).

To run BRAMS 4.0 in the standard mode as a atmospheric model, i.e. Not using CATT or TEB, the user need to set two variables, in the sections

```
$CATT_INFO and $TEB_INFO, to deactivate these emission models:
$CATT_INFO
CATT = 0, ! 1-CATT environmental model activated 0-off
…
$END
$TEB_INFO
TEB = 0, ! 1-TEB activated 0-off
…
$END
```

The output files produced by BRAMS forecast phase are History and Analysis files. History files contain all information required to restart the model. Analysis files contain the output of the forecast. Variables that control Analysis and History file paths, prefix, and frequency are shown bellow (all at namelist $MODEL_FILE_INFO):

```
IOUTPUT  =  2,             ! 0-no files, 1-save ASCII, 2-save binary
HFILOUT  = './H/hist',
AFILOUT  = './A/anal',
ICLOBBER = 1,                ! 0=stop if files exist, 1=overwite files
IHISTDEL = 1,                ! 0=keep all hist files, 1=delete previous
FRQHIS   = 21600.,         ! History file frequency
FRQANL   = 10800.,         ! Analysis file frequency
```

The variable *IOUTPUT* defines the type of output file. History file path and prefix are specified by *HFILOUT*. The equivalent variable for Analysis is *AFILOUT*. Variables *ICLOBBER* and *IHISTDEL* should be used to avoid overwriting files from previous executions. Variables *FRQHIS* and *FRQANL* contain output frequency (in seconds of simulation time).

1.6.5 Post processing and picture generation

The fifth step of the script is to do post-processing and picture generation, both performed by script *run.sh* at directory RAMSPOST.

Program *ramspost50.x*, at the RAMSPOST directory is used to post-process BRAMS forecast files. It takes as input the namelist file *ramspost.inp* at the same directory, and produces output files *result_g1.ctl* and *result_g1.gra*, at the same directory. The program just selects fields from analysis files and prepares them for grads visualization.

File *result_g1.ctl* describes the contents of *result_g1.gra* in a format suited for grads. The grads visualizing tool can be started, ingesting the .ctl file and display available fields at user's discretion.

To certify the installation, the *run.sh* script invokes grads with the script *toptmp.gs*, that just display and prints topography and temperature at files *topo.gif* and *temp.gif*, all at the RAMSPOST directory.

1.7. Configuring BRAMS

BRAMS possess just too many configuring switches, as shown by the size of RAMSIN. That is a design constraint: BRAMS should be very flexible to simplify research. Consequently, it is beyond the scope of this document to show how to modify all the switches. We restrict ourselves to the most common cases: changing initial date and forecast time extension, forecast resolution and geographical area setting.

1.7.1 Changing initial date and forecast time length

First step is to download into directory *dprep* CPTEC global files for the desired date and forecast length. CPTEC global files should be collected with anonymous *ftp* from the site www.cptec.inpe.br (IP address 150.163.141.140). At this site, go to directory pub/produtos/prod_eta/pnt_sx6/GAMRAMS126/.

Collect files with prefix *GAMRAMS* and suffix *fct.T126L28.grb*, such as `GAMRAMS2005102500200510310 6P.fct.T126L28.grb`. Filename syntax is *GAMRAMS<begin date><current date>P.fct.T126L28.grb*, where <*date*> has the format YYYYMMDDHH (year YYYY, month MM, day DD and hour HH). The <*begin date*> represents the starting time of the global model forecast; while <*current date*> represents data output time. Both dates stand for the state of the atmosphere instantly at the precise date. Consequently, file `GAMRAMS2005102500200510310 6P.fct.T126L28.grb` contains the forecasted state of the atmosphere as of 06 hours (GMT) of October 31st 2005, obtained from a global model run starting with the state of the atmosphere as of 00 hours (GMT) of October 25th 2005.

Be sure to collect all files required for the local area forecast. These encompass all files from starting date to end date. Intermediate files should be available every 6 hours, to establish boundary conditions.

Second step is to specify the forecast initial date. On namelist $MODEL_GRID at RAMSIN, change variables:

```
IMONTH1 = 10,
IDATE1 = 25,
IYEAR1 = 2005,
ITIME1 = 00,
```

to the desired forecast starting date.

Third step is to specify forecast time length (and time unit) by setting appropriate values for variables *TIMMAX* and *TIMEUNIT* at the same namelist.

Note that the forecast must start at the date and hour indicated by some GAMRAMS filename. For example, if you download files:

```
GAMRAMS2005102500200510310 0P.fct.T126L28.grb
GAMRAMS2005102500200510310 6P.fct.T126L28.grb
GAMRAMS2005102500200510311 2P.fct.T126L28.grb
GAMRAMS2005102500200510311 8P.fct.T126L28.grb
GAMRAMS2005102500200511010 0P.fct.T126L28.grb
```

then forecast could start with variable *ITIME1* equals to one of 00, 06, 12 or 18.

The most recent GAMRAMS file must have <*current time*> either equal or superior to the date defined by the sum of the starting date of the forecast with the value of the

TIMMAX variable. To the GAMRAMS files listed above, if the start of simulation is set to 00 hour of October 25th of 2005, the *TIMMAX* and *TIMEUNIT* could be set up to 24 and H respectively. If the starting hour is set to 12, the *TIMMAX* could be set up to 12.

Once these dates are set, just run all the five script steps. Make sure all three RAMSIN are correctly modified.

1.7.2Changing model resolution

Limited area size and resolution are defined by four variables at $MODEL_GRID namelist:

```
NNXP = 35,78,34, ! Number of x gridpoints
NNYP = 34,82,34, ! Number of y gridpoints
DELTAX = 112000.,
DELTAY = 112000., ! X and Y grid spacing
```

Observe that while *DELTAX* and *DELTAY* (both Fortran real variables) define model resolution (in meters), the length of the limited area at each dimension is the product of the number of grid intervals (which is the number of grid points minus one) by the grid spacing at each direction. Usually the x axis represents longitudes while the y axis stands for latitudes.

Note that the *NNXP* and *NNYP* variables (both Fortran integer variables) have a three numbers sequence. As previously stated, the second and the third numbers define a three nested grid system, with increasing grid refinement. In the first time user version, only the first term should be changed, since variable *NGRIDS* is set to 1.

Changing model resolution comes with a price. As expected, there is an increase in execution time due to the increase in domain points. But there is another factor that increases execution time: the value of time step. Whenever resolution increases (decreasing *DELTAX* or *DELTAY*) time step should be decreased to maintain numerical stability. Time step is the value of variable *DTLONG* (Fortran real, in seconds) at the same namelist. A trial and error procedure to establish the most adequate time step value is strongly recommended.

1.7.3Changing limited area location

Forecasted area is defined by its size and by the location of its center. The center of the grid is defined by the namelist $MODEL_GRID variables:

```
CENTLAT = -23.0, -23.0, -23.2,
CENTLON = -55.0, -52.5, -47.0,
```

CENTLAT and *CENTLON* (both Fortran real variables) respectively define the latitude and the longitude to the grid's center, in degrees from the Equator and Greenwich. Again, both variables have a three numbers sequence for a nested grid system. In the first time user version only the first term can be changed.

1.7.4 Configuring the Ramspost

The rampost.inp is the file that governs the ramspost50.x run. It is a namelist file that contains all the main definitions for post processing the analysis files and to generate the GRADS' input files.

The first variable to define is the FPREFIX, that defines location and prefix of the analysis files (BRAMS'output). In BRAMS First Time User version, the FPREFIX has the following value:

```
FPREFIX = '../A/anal-A-',
```

With this definition, the ramspost50,x will work over all the files that have the prefix 'anal-A-' stored in ./A directory. If the analysis files of your interest have other name or they are stored in other directory, change the FPREFIX value.

It may be the case that two BRAMS runs with distinct configurations (dates, domain, etc) store analysis files (BRAMS' output) at the same directory. In such a case, it is important to change the prefix of the analysis files in each run. This prevents the RAMSPOST to interpolate the results from different simulations and generate strange results. Remember: change the analysis prefix in all RAMSIN files and in ramspost.inp file.

The variables NVP and VP indicate the amount of the variables to be visualized and which are the variable. See this example from the First Time User:

```
NVP = 20,
VP  =    'totpcp',
         'sst',
         'topo',
         'vtype',
         'stext',
         'smoist',
         'w',
         'u',
         'v',
         'tempc',
         'theta',
         'rh',
         'rv',
         'pcprate',
         'acccon',
         'h',
         'le',
         'rlong',
         'rshort',
         'rlongup',
```

In this case, twenty variables were defined to visualize. Each variable has a physical mean and it is possible to know the meaning of each one with a help of the rampost50.x output file with the .ctl extension, like is shown bellow:

```
vars   23
totpcp       0 99    - RAMS : total resolved precip                  [mm liq  ]
sst          0 99    - RAMS : water temperature                      [C       ]
topo         0 99    - RAMS : topo                                   [m       ]
vtype1       0 99    - RAMS : vegetation class: patch # 1            [#       ]
vtype2       0 99    - RAMS : vegetation class: patch # 2            [#       ]
stext1       9 99    - RAMS : soil twxture: patch # 1                [        ]
stext2       9 99    - RAMS : soil twxture: patch # 2                [        ]
smoist1      9 99    - RAMS : soil moisture: patch # 1               [m3/m3   ]
smoist2      9 99    - RAMS : soil moisture: patch # 2               [m3/m3   ]
w           30 99    - RAMS : w                                      [m/s     ]
u           30 99    - RAMS : u                                      [m/s     ]
v           30 99    - RAMS : v                                      [m/s     ]
tempc       30 99    - RAMS : temperature                            [C       ]
theta       30 99    - RAMS : potential temp                         [K       ]
rh          30 99    - RAMS : relative humidity                      [pct     ]
rv          30 99    - RAMS : vapor mix ratio                        [g/kg    ]
pcprate      0 99    - RAMS : resolved precip rate                   [mm/hr   ]
acccon       0 99    - RAMS : accum convective pcp                   [mm      ]
h            0 99    - RAMS : sfc sens heat flx                      [W/m2    ]
le           0 99    - RAMS : sfc lat heat flx                       [W/m2    ]
rlong        0 99    - RAMS : rlong                                  [W/m2    ]
rshort       0 99    - RAMS : rshort                                 [W/m2    ]
rlongup      0 99    - RAMS : rlongup                                [W/m2    ]
endvars
```

Some variable names like vtype and stext generate more than one field to be visualized. That is why rampost changes the initial variable count (20 at rampost.inp) into (23 at .ctl file).

There are many variables that can be selected for visualization. The variable name follows the pattern defined by RAMS/HYPACT Evaluation and Visualization Utilities (REVU).

To visualize the output files of ramspost50.x (result_g1.ctl and result_g1.gra), there is a script file called toptmp.gs than control the GRADS' run:

```
'open result_g1.ctl'
'set mpdset ./mres'
'enable print topo'
'set gxout shaded'
'd topo'
'draw title Topography'
'run ./cbarn'
'print'
'disable print'
'!./gxgif -r -i topo'
'clear'
'set mpdset ./mres'
'enable print temp'
'set gxout contour'
'd tempc'
'draw title Temperature 12H(GMT) 24/01/2005'
'print'
'disable print'
```

```
'!./gxgif -r -i temp'
'quit'
```

The commands “d topo” and “d tempc” makes the GRADS draw the topography and the temperature field. The command draw title writes the title. The command `'!./gxgif -r -i topo'` and `!./gxgif -r -i temp'` generate the gif files showed in the begin of this document.

To visualize other variables from a same simulation, there are two situations:

- First: the desired variable is in ramspost.inp list. In this case, change the toptmp.gs file with the desired variable name. Execute the run.sh script again.
- Second: the desired variable is not in ramspost.inp list. Include the desired variable name in ramspost.inp and change the toptmp.gs file. Execute the run.sh script again.

For more informations see the RAMSPOST User Guide:

`http://www.cptec.inpe.br/brams/ramspost.shtml`

APENDIX A – Attribution ShareAlike 2.5 Brasil commons license (summary)

Attribution-ShareAlike 2.5

You are free:

- to copy, distribute, display, and perform the work
- to make derivative works
- to make commercial use of the work

Under the following conditions:

Attribution. You must attribute the work in the manner specified by the author or licensor.

Share Alike. If you alter, transform, or build upon this work, you may distribute the resulting work only under a license identical to this one.

- For any reuse or distribution, you must make clear to others the license terms of this work.
- Any of these conditions can be waived if you get permission from the copyright holder.

Your fair use and other rights are in no way affected by the above.

This is a human-readable summary of the Legal Code (Full license)

APENDIX B – Attribution ShareAlike 2.5 Brasil commons license (full license)

Attribution-ShareAlike 2.5

CREATIVE COMMONS CORPORATION IS NOT A LAW FIRM AND DOES NOT PROVIDE LEGAL SERVICES. DISTRIBUTION OF THIS LICENSE DOES NOT CREATE AN ATTORNEY-CLIENT RELATIONSHIP. CREATIVE COMMONS PROVIDES THIS INFORMATION ON AN "AS-IS" BASIS. CREATIVE COMMONS MAKES NO WARRANTIES REGARDING THE INFORMATION PROVIDED, AND DISCLAIMS LIABILITY FOR DAMAGES RESULTING FROM ITS USE.

License

THE WORK (AS DEFINED BELOW) IS PROVIDED UNDER THE TERMS OF THIS CREATIVE COMMONS PUBLIC LICENSE ("CCPL" OR "LICENSE"). THE WORK IS PROTECTED BY COPYRIGHT AND/OR OTHER APPLICABLE LAW. ANY USE OF THE WORK OTHER THAN AS AUTHORIZED UNDER THIS LICENSE OR COPYRIGHT LAW IS PROHIBITED.

BY EXERCISING ANY RIGHTS TO THE WORK PROVIDED HERE, YOU ACCEPT AND AGREE TO BE BOUND BY THE TERMS OF THIS LICENSE. THE LICENSOR GRANTS YOU THE RIGHTS CONTAINED HERE IN CONSIDERATION OF YOUR ACCEPTANCE OF SUCH TERMS AND CONDITIONS.

1. Definitions

a. **"Collective Work"** means a work, such as a periodical issue, anthology or encyclopedia, in which the Work in its entirety in unmodified form, along with a number of other contributions, constituting separate and independent works in themselves, are assembled into a collective whole. A work that constitutes a Collective Work will not be considered a Derivative Work (as defined below) for the purposes of this License.

b. **"Derivative Work"** means a work based upon the Work or upon the Work and other pre-existing works, such as a translation, musical arrangement, dramatization, fictionalization, motion picture version, sound recording, art reproduction, abridgment, condensation, or any other form in which the Work may be recast, transformed, or adapted, except that a work that constitutes a Collective Work will not be considered a Derivative Work for the purpose of this License. For the avoidance of doubt, where the Work is a musical composition or sound recording, the synchronization of the Work in timed-relation with a moving image ("synching") will be considered a Derivative Work for the purpose of this License.

c. **"Licensor"** means the individual or entity that offers the Work under the terms of this License.

d. **"Original Author"** means the individual or entity who created the Work.

e. **"Work"** means the copyrightable work of authorship offered under the terms of this License.

f. **"You"** means an individual or entity exercising rights under this License who has not previously violated the terms of this License with respect to the Work, or who has received express permission from the Licensor to exercise rights under this License despite a previous violation.

g. **"License Elements"** means the following high-level license attributes as selected by Licensor and indicated in the title of this License: Attribution, ShareAlike.

2. Fair Use Rights. Nothing in this license is intended to reduce, limit, or restrict any rights arising from fair use, first sale or other limitations on the exclusive rights of the copyright owner under copyright law or other applicable laws.

3. License Grant. Subject to the terms and conditions of this License, Licensor hereby grants You a worldwide, royalty-free, non-exclusive, perpetual (for the duration of the applicable copyright) license to exercise the rights in the Work as stated below:

a. to reproduce the Work, to incorporate the Work into one or more Collective Works, and to reproduce the Work as incorporated in the Collective Works;

b. to create and reproduce Derivative Works;

c. to distribute copies or phonorecords of, display publicly, perform publicly, and perform publicly by means of a digital audio transmission the Work including as incorporated in Collective Works;

d. to distribute copies or phonorecords of, display publicly, perform publicly, and perform publicly by means of a digital audio transmission Derivative Works.

e. For the avoidance of doubt, where the work is a musical composition:

i. **Performance Royalties Under Blanket Licenses**. Licensor waives the exclusive right to collect, whether individually or via a performance rights society (e.g. ASCAP, BMI, SESAC), royalties for the public performance or public digital performance (e.g. webcast) of the Work.

ii. **Mechanical Rights and Statutory Royalties**. Licensor waives the exclusive right to collect, whether individually or via a music rights society or designated agent (e.g. Harry Fox Agency), royalties for any phonorecord You create from the Work ("cover version") and distribute, subject to the compulsory license created by 17 USC Section 115 of the US Copyright Act (or the equivalent in other jurisdictions).

f. **Webcasting Rights and Statutory Royalties**. For the avoidance of doubt, where the Work is a sound recording, Licensor waives the exclusive right to collect, whether individually or via a performance-rights society (e.g. SoundExchange), royalties for the public digital performance (e.g. webcast) of the Work, subject to the compulsory license created by 17 USC Section 114 of the US Copyright Act (or the equivalent in other jurisdictions).

The above rights may be exercised in all media and formats whether now known or hereafter devised. The above rights include the right to make such modifications as are

technically necessary to exercise the rights in other media and formats. All rights not expressly granted by Licensor are hereby reserved.

4. Restrictions. The license granted in Section 3 above is expressly made subject to and limited by the following restrictions:

a. You may distribute, publicly display, publicly perform, or publicly digitally perform the Work only under the terms of this License, and You must include a copy of, or the Uniform Resource Identifier for, this License with every copy or phonorecord of the Work You distribute, publicly display, publicly perform, or publicly digitally perform. You may not offer or impose any terms on the Work that alter or restrict the terms of this License or the recipients' exercise of the rights granted hereunder. You may not sublicense the Work. You must keep intact all notices that refer to this License and to the disclaimer of warranties. You may not distribute, publicly display, publicly perform, or publicly digitally perform the Work with any technological measures that control access or use of the Work in a manner inconsistent with the terms of this License Agreement. The above applies to the Work as incorporated in a Collective Work, but this does not require the Collective Work apart from the Work itself to be made subject to the terms of this License. If You create a Collective Work, upon notice from any Licensor You must, to the extent practicable, remove from the Collective Work any credit as required by clause 4(c), as requested. If You create a Derivative Work, upon notice from any Licensor You must, to the extent practicable, remove from the Derivative Work any credit as required by clause 4(c), as requested.

b. You may distribute, publicly display, publicly perform, or publicly digitally perform a Derivative Work only under the terms of this License, a later version of this License with the same License Elements as this License, or a Creative Commons iCommons license that contains the same License Elements as this License (e.g. Attribution-ShareAlike 2.5 Japan). You must include a copy of, or the Uniform Resource Identifier for, this License or other license specified in the previous sentence with every copy or phonorecord of each Derivative Work You distribute, publicly display, publicly perform, or publicly digitally perform. You may not offer or impose any terms on the Derivative Works that alter or restrict the terms of this License or the recipients' exercise of the rights granted hereunder, and You must keep intact all notices that refer to this License and to the disclaimer of warranties. You may not distribute, publicly display, publicly perform, or publicly digitally perform the Derivative Work with any technological measures that control access or use of the Work in a manner inconsistent with the terms of this License Agreement. The above applies to the Derivative Work as incorporated in a Collective Work, but this does not require the Collective Work apart from the Derivative Work itself to be made subject to the terms of this License.

c. If you distribute, publicly display, publicly perform, or publicly digitally perform the Work or any Derivative Works or Collective Works, You must keep intact all copyright notices for the Work and provide, reasonable to the medium or means You are utilizing: (i) the name of the Original Author (or pseudonym, if applicable) if supplied, and/or (ii) if the Original Author and/or Licensor designate another party or parties (e.g. a sponsor institute, publishing entity, journal) for attribution in Licensor's copyright notice, terms of service or by other reasonable

means, the name of such party or parties; the title of the Work if supplied; to the extent reasonably practicable, the Uniform Resource Identifier, if any, that Licensor specifies to be associated with the Work, unless such URI does not refer to the copyright notice or licensing information for the Work; and in the case of a Derivative Work, a credit identifying the use of the Work in the Derivative Work (e.g., "French translation of the Work by Original Author," or "Screenplay based on original Work by Original Author"). Such credit may be implemented in any reasonable manner; provided, however, that in the case of a Derivative Work or Collective Work, at a minimum such credit will appear where any other comparable authorship credit appears and in a manner at least as prominent as such other comparable authorship credit.

5. Representations, Warranties and Disclaimer

UNLESS OTHERWISE AGREED TO BY THE PARTIES IN WRITING, LICENSOR OFFERS THE WORK AS-IS AND MAKES NO REPRESENTATIONS OR WARRANTIES OF ANY KIND CONCERNING THE MATERIALS, EXPRESS, IMPLIED, STATUTORY OR OTHERWISE, INCLUDING, WITHOUT LIMITATION, WARRANTIES OF TITLE, MERCHANTIBILITY, FITNESS FOR A PARTICULAR PURPOSE, NONINFRINGEMENT, OR THE ABSENCE OF LATENT OR OTHER DEFECTS, ACCURACY, OR THE PRESENCE OF ABSENCE OF ERRORS, WHETHER OR NOT DISCOVERABLE. SOME JURISDICTIONS DO NOT ALLOW THE EXCLUSION OF IMPLIED WARRANTIES, SO SUCH EXCLUSION MAY NOT APPLY TO YOU.

6. Limitation on Liability. EXCEPT TO THE EXTENT REQUIRED BY APPLICABLE LAW, IN NO EVENT WILL LICENSOR BE LIABLE TO YOU ON ANY LEGAL THEORY FOR ANY SPECIAL, INCIDENTAL, CONSEQUENTIAL, PUNITIVE OR EXEMPLARY DAMAGES ARISING OUT OF THIS LICENSE OR THE USE OF THE WORK, EVEN IF LICENSOR HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

7. Termination

a. This License and the rights granted hereunder will terminate automatically upon any breach by You of the terms of this License. Individuals or entities who have received Derivative Works or Collective Works from You under this License, however, will not have their licenses terminated provided such individuals or entities remain in full compliance with those licenses. Sections 1, 2, 5, 6, 7, and 8 will survive any termination of this License.

b. Subject to the above terms and conditions, the license granted here is perpetual (for the duration of the applicable copyright in the Work). Notwithstanding the above, Licensor reserves the right to release the Work under different license terms or to stop distributing the Work at any time; provided, however that any such election will not serve to withdraw this License (or any other license that has been, or is required to be, granted under the terms of this License), and this License will continue in full force and effect unless terminated as stated above.

8. Miscellaneous

a. Each time You distribute or publicly digitally perform the Work or a Collective Work, the Licensor offers to the recipient a license to the Work on the same terms and conditions as the license granted to You under this License.

b. Each time You distribute or publicly digitally perform a Derivative Work, Licensor offers to the recipient a license to the original Work on the same terms and conditions as the license granted to You under this License.

c. If any provision of this License is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this License, and without further action by the parties to this agreement, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.

d. No term or provision of this License shall be deemed waived and no breach consented to unless such waiver or consent shall be in writing and signed by the party to be charged with such waiver or consent.

e. This License constitutes the entire agreement between the parties with respect to the Work licensed here. There are no understandings, agreements or representations with respect to the Work not specified here. Licensor shall not be bound by any additional provisions that may appear in any communication from You. This License may not be modified without the mutual written agreement of the Licensor and You.

Creative Commons is not a party to this License, and makes no warranty whatsoever in connection with the Work. Creative Commons will not be liable to You or any party on any legal theory for any damages whatsoever, including without limitation any general, special, incidental or consequential damages arising in connection to this license. Notwithstanding the foregoing two (2) sentences, if Creative Commons has expressly identified itself as the Licensor hereunder, it shall have all rights and obligations of Licensor.

Except for the limited purpose of indicating to the public that the Work is licensed under the CCPL, neither party will use the trademark "Creative Commons" or any related trademark or logo of Creative Commons without the prior written consent of Creative Commons. Any permitted use will be in compliance with Creative Commons' then-current trademark usage guidelines, as may be published on its website or otherwise made available upon request from time to time.

Creative Commons may be contacted at http://creativecommons.org/.

APENDIX C – Free Software Needs Free Documentation

The following article was written by Richard Stallman, founder of the GNU Project.

The biggest deficiency in the free software community today is not in the software--it is the lack of good free documentation that we can include with the free software. Many of our most important programs do not come with free reference manuals and free introductory texts. Documentation is an essential part of any software package; when an important free software package does not come with a free manual and a free tutorial, that is a major gap. We have many such gaps today.

Consider Perl, for instance. The tutorial manuals that people normally use are non-free. How did this come about? Because the authors of those manuals published them with restrictive terms--no copying, no modification, source files not available--which exclude them from the free software world.

That wasn't the first time this sort of thing happened, and it was far from the last. Many times we have heard a GNU user eagerly describe a manual that he is writing, his intended contribution to the community, only to learn that he had ruined everything by signing a publication contract to make it non-free.

Free documentation, like free software, is a matter of freedom, not price. The problem with the non-free manual is not that publishers charge a price for printed copies--that in itself is fine. (The Free Software Foundation sells printed copies of manuals, too.) The problem is the restrictions on the use of the manual. Free manuals are available in source code form, and give you permission to copy and modify. Non-free manuals do not allow this.

The criteria of freedom for a free manual are roughly the same as for free software. Redistribution (including the normal kinds of commercial redistribution) must be permitted, so that the manual can accompany every copy of the program, both on-line and on paper.

Permission for modification of the technical content is crucial too. When people modify the software, adding or changing features, if they are conscientious they will change the manual too--so they can provide accurate and clear documentation for the modified program. A manual that leaves you no choice but to write a new manual to document a changed version of the program is not really available to our community.

Some kinds of limits on the way modification is handled are acceptable. For example, requirements to preserve the original author's copyright notice, the distribution terms, or the list of authors, are ok. It is also no problem to require modified versions to include notice that they were modified. Even entire sections that may not be deleted or changed are acceptable, as long as they deal with nontechnical topics (like this one). These kinds of restrictions are acceptable because they don't obstruct the community's normal use of the manual.

However, it must be possible to modify all the *technical* content of the manual, and then distribute the result in all the usual media, through all the usual channels. Otherwise,

the restrictions obstruct the use of the manual, it is not free, and we need another manual to replace it.

Please spread the word about this issue. Our community continues to lose manuals to proprietary publishing. If we spread the word that free software needs free reference manuals and free tutorials, perhaps the next person who wants to contribute by writing documentation will realize, before it is too late, that only free manuals contribute to the free software community.

If you are writing documentation, please insist on publishing it under the GNU Free Documentation License or another free documentation license. Remember that this decision requires your approval--you don't have to let the publisher decide. Some commercial publishers will use a free license if you insist, but they will not propose the option; it is up to you to raise the issue and say firmly that this is what you want. If the publisher you are dealing with refuses, please try other publishers. If you're not sure whether a proposed license is free, write to `licensing at gnu.org`.

You can encourage commercial publishers to sell more free, copylefted manuals and tutorials by buying them, and particularly by buying copies from the publishers that paid for their writing or for major improvements. Meanwhile, try to avoid buying non-free documentation at all. Check the distribution terms of a manual before you buy it, and insist that whoever seeks your business must respect your freedom. Check the history of the book, and try reward the publishers that have paid or pay the authors to work on it.

The Free Software Foundation maintains a list of free documentation published by other publishers:

- http://www.fsf.org/doc/other-free-books.html