The Abdus Salam
International Centre for Theoretical Physics
Strada Costiera, 11 I - 34151 Trieste, Italy
Earth System Physics Section - ESP

Regional Climatic Model RegCM User's Guide Version 4.1 Trieste, Italy - May 2011

Filippo Giorgi, Nellie Elguindi,
Stefano Cozzini and Graziano Giuliani

Contents

Chapter 1

Release Notes

RegCM-4.1 is a new step in recoding the RegCM3 model after the effort put into the RegCM4.0 version. The code base now is actively developed by a community of developers internal and external to ICTP, and this work is merged on the eforge site on e-science-lab.org site.

The main new technical features of the code are summarized in the following points

- Dynamic memory allocation
- F95 programming language
- Module approach
- Unique makefile for the whole code from pre-processing to post-processing
- netCDF I/O format from model components following the CF-1.4 standard

The model code is in Fortran 90 ANSI standard with some language extensions of Fortran 2003 implemented in all the supported compilers. The development is done on Linux boxes, and the model is known to run on Oracle Solaris^{TM}platforms, IBM AIX^{TM}platforms, MacOS^{TM}platforms. No porting effort has been done towards non Unix-like Operating Systems. We will for this User Guide assume that the reference platform is a recent Linux distributon with a `bash` shell. Typographical convention is the following:

Table 1.1: Conventions

`$>`	normal shell prompt
`#>`	root shell prompt
`$SHELL_VARIABLE`	a shell variable

Any shell variable is supposed to be set by the User with the following example syntax:

```
$> export REGCM_ROOT="/home/graziano/RegCM4.1"
```

Hope you will find this document useful. Any error found belongs to me and can be reported to be corrected in future revisions. Enjoy.

Chapter 2

Obtain the model

2.1 Simple Model User

A packed archive file with the model code can be downloaded from:

`http://eforge.escience-lab.org/gf/project/regcm/frs/RegCM-4.1.tar.gz`

and it can be later on decompressed and unpacked using:

```
$> tar -zxvf RegCM-4.1.tar.gz
```

2.2 Model Developer

If you plan to become a model developer, source code can be obtained via svn. The RegCM team strongly encouragethe contributing developers to enroll on the eforge site to always be up to date and to check on-line all the news of the package.

https://eforge.escience-lab.org/gf/project/regcm

The correct procedure is first to register on the e-forge site, then ask the ICTP scientific team head Filippo Giorgi to be enrolled as a model developer. After being officially granted the status, you will gain access to the model subversion repository.

Check that **Subversion** software is installed on your machine typing the following command:

```
$> svn --version
```

If your system answers `command not found`, refer to your System Administrator or software installation manual of your OS to install the subversion software. As an example, on Scientific Linux the command to install it as root is:

```
#> yum install subversion
```

If Subversion is installed, just type the following command:

```
$> svn checkout https://eforge.escience-lab.org/svn/regcm/tags/RegCM-4.1
```

Chapter 3

Installing procedure

Whatever method is chosen to download the code, we assume that you have now on your working directory a new directory, named `RegCM-4.1`. That directory will be for the rest of this guide referred as `$REGCM_ROOT` .

All the operations to build the model binaries will be performed in this directory.

3.1 Software requirements

In order to configure and install the RegCM code, the following software are needed:

1. Python 2 language interpreter
2. GNU Make program
3. Fortran 90 compiler
4. netCDF *Rew and Davis* (1990) Format I/O library compiled with the above compiler. Source code can be found from
   ftp://ftp.unidata.ucar.edu/pub/netcdf/netcdf.tar.gz
   Note that current version 4.1.2 is dependent on HDF5 1.8.6.

Optional requirements strongly suggested are :

1. GNU `patch` program if CLM option is activated.
2. MPI2 Message Passing Library compiled with the above fortran compiler for parallel runs using multiple core single machines or cluster of machines. Source code for the implementation code was tested with can be obtained at:
   http://www.open-mpi.org/software/ompi/v1.4/downloads
3. HDF5 Format I/O Library compiled with the above fortran compiler to enable netCDF V4 features. Source code can be obtained at:
   http://www.hdfgroup.org/ftp/HDF5/current/src

4. NCO netCDF Operators for manging netCDF file. Most Linux distribution have this already packed, and you should refer to your System Administrator or OS Software Installation manual to obtain it. Source code is at:
   http://nco.sourceforge.net/src

5. CDO Climatic data Operators for managing netCDF file. Most Linux distribution have this already packed, and you should refer to your System Administrator or OS Software Installation manual to obtain it. Source code is at:
   https://code.zmaw.de/projects/cdo/files

6. A Scientific Plotting and Data Analysis Software such as:

   - IGES GrADS 2.0 Graphical Analysis and Display System. Convenient helpers are packed in RegCM to use GrADS with RegCM netCDF output files. Binaries and source code can be obtained from **http://www.iges.org/grads/downloads.html**
   - NCL, NCAR CISL Command Language. The NCL can read netCDF output files, and sample scripts can be found in the *Tools/Scripts/NCL* directory. Binaries and source code can be obtained from **http://www.ncl.ucar.edu**

7. A quick viewer for netCDF files like NcView:
   http://meteora.ucsd.edu/ pierce/ncview_home_page.html

An example session of installation of basic software needed to compile the RegCM model is detailed in chapter 9.

3.2 Configuring build

The RegCM Version 4.1 is configured by a python2 script, which will select and edit for you sample configuration files for the supported architectures. These files are kept in the `Arch` directory under `$REGCM_ROOT`.

Currently tested and supported configurations (OS/Compiler) are:

1. Linux with GNU gfortran compiler version ≥ 4.4
2. Linux with Intel^{TM}ifort compiler version ≥ 10.0
3. Linux with Portland^{TM}pgf90 compiler version ≥ 9.3
4. Mac OsX^{TM}with g95 compiler
5. IBM AIX^{TM}with xlf compiler
6. Oracle Solaris^{TM}with Oracle Solaris Studio^{TM}compiler ≥ 8.3

The first step is to change working directory to `$REGCM_ROOT` and run the `configure` script:

```
$> cd $REGCM_ROOT
$> ./configure
        ######################################
        # Welcome to the RegCM configuration! #
        ######################################

  This script will help you to setup the RegCM distribution
  with some simple questions. Default options [between square brackets]
  will be assumed if you don't select valid options.
```

The script interactively asks you a series of questions, divided into three distinct groups:

- General path and software requirements
- Model configuration
- Choice of Fortran compiler

3.2.1 General path and software requirements

1. The location of `$REGCM_ROOT`.

   ```
   ****  Please enter the path to your RegCM distribution:
   [ /home/regcm/RegCM4.1 ]
   regcm_root =
   ```

   This is because you may be running the script from a different directory. You probably are not, so just press enter.

2. Built binaries destination directory.

   ```
   ****  Please enter the path where the RegCM binaries will be stored:
   [ /home/regcm/RegCM4.1/Bin ]
   ```

   If you are the only one using the model on the machine, you may safely accept the default choice pressing enter.

3. netCDF installation path. The script tries to detect the netCDF installation automatically using the `nc-config` program, searching standard directories and using the `NETCDF` environment variable. If successful, the message

   ```
   netCDF library found...
   ```

   should appear. Otherwise you will be prompted the following question:

   ```
   Unable to find a working netCDF. Please input a valid path...
   For netCDF library =
   ```

   The answer should be the directory where on your system the file `libnetcdf.a` is stored. Next the file will ask:

   ```
   For netCDF include files =
   ```

The answer should be the directory where on your system the file `netcdf.mod` is stored.

4. OPTIONAL - If version 4 of netCDF library is detected, but the script is unable to use the `nc-config` program, and autodetection using standard paths or the HDF5 environment variable fails, you may be prompted also for location of HDF5 library

   ```
   Unable to find a working HDF5. Please input a valid path...
   For HDF5 library =
   ```

   The answer should be the directory where on your system the file `libhdf5.a` is stored.

5. OPTIONAL - If version 4 of netCDF library is detected, the script will ask if to enable netCDF file compression filter.

   ```
   ****  Would you like to enable netCDF v4 compression? (0/1)
   [0]
   zlib =
   ```

   As the compression causes a run time penalty, it is not enabled by default. If you are concerned by disk space, this option may help significantly reduce storage usage by model output, especially if the chemistry option is enabled. To enable it enter `1`, otherwise just accept default option pressing enter.

3.2.2 Model configuration

The configure script goes to the second phase.

```
!!!!  The following options are all documented in detail
      on the eForge wiki page and in the RegCM Users' Guide,
      so please check there for more information.
```

1. Enable debug

   ```
   ****  Do you want a debug - 0 or a production - 1  binary?
         [1]
         dbg =
   ```

   If enabled, the model will be compiled using debug flags for the compiler, which will allow the use of a debugger such as gdb. More diagnostics will also be generated during model run. To enable debug mode, enter `0`. The default is to build production binaries with all optimization flags turned on.

2. Parallel code using MPI library

   ```
   ****  Do you want a serial - 0 or MPI-parallel - 1 binary?
         [1]
         mpi =
   ```

The model is coded to use an MPI2 library to run in parallel mode using multiple cores/processors or run on a cluster. To enable this option you will need to have installed on your system this library. Note that even if built using MPI, the model can be still executed in serial mode using at run time just one processor. The penalty time in this case is less than 1% of the run time of a serial built executable. The RegCM team strongly suggest to build MPI enabled model, as the serial option will be dropped in future releases. To enable serial build, enter `0`, otherwise accept the default choice pressing enter. [1]

3. BAND option

```
****  Do you want to run the tropical band version, BAND - 1 or not - 0?
      [0]
      BAND =
```

This option builds a special version of the model capable of running an experiment with a spatial domain configured as a full circular equatorial band around earth. This is documented in the *Giorgi* (2011). The default is to not enable this feature, i.e. to run the model on a limited area not going round the whole earth. If you are planning to run such a simulation, enter `1`, otherwise accept the default choice pressing enter. For the scope of the tutorial test run in chapter 5, use the default option.

4. CLM option

```
****  Do you want to enable CLM  - 1 or not - 0?
      [0]
      CLM =
```

This option switches off the default Land model of RegCM (derived from BATS1e), and enables the use of the Community Land Model V3.5 inside RegCM. The default is to use the RegCM BATS Land Model. If you are planning to run a simulation using CLM, enter `1`, otherwise accept the default choice pressing enter. [2] For the scope of the tutorial test run in chapter 5, use the default option.

3.2.3 Choice of Fortran compiler

The `configure` script now prompts the user for selecting a fortran compiler. Depending on your OS/compiler combination, select a supported one in the list , or ask for a generic configuration.

```
****  The following compiler/architecture combinations
      are tested and known to work, please choose one:

            1. GNU Fortran v. 4.4 (Linux x86-64)
            2. Intel Fortran v. 10 or 11 (Linux x86-64)
            3. PGI Fortran v. 9 (Linux x86-64)
```

[1] The serial version of the model is deprecated, and support for this non MPI version will be dropped in future model releases.

[2] The CLM option needs the GNU patch program to be installed.

```
            4. IBM Xlf Compiler (generic AIX)
            5. IBM Xlf Compiler (Cineca SP6)
            6. Sun Ceres Fortran 95 v 8.3 r2008/01/28 (Linux x86-64)
            7. GNU+g95 Fortran compiler (Linux x86-64)
            8. Other

      compiler =
```

If your OS/compiler is not listed, enter `8` for a generic system. You are required afterwards to edit the `Makefile.inc` file in the `$REGCM_ROOT` directory to configure you build. Otherwise, select one in the list. The `configure` script should now create a `Makefile.inc` file inside `$REGCM_ROOT` to be used to compile the RegCM model. You may now want to double check the `Makefile.inc` file, or accept what we have arranged for you.

3.3 Build the model executables

Now that everything is hopefully configured, you may use the `make` program to build executables. You have multiple targets for the build, but at this stage you may want to

```
$> make all
```

This target will builds all model parts. Other allowed targets are:

1. terrain : Just build the terrain program
2. icbc : Just build the icbc programs
3. clm2rcm : If CLM option selected, just build the clm2rcm program
4. regcm : Build the model executable in Main directory
5. postproc : Build postprocessing programs in PostProc
6. clean : clean up the build directory for objects

The compilation is started in the whole model tree (PreProc, Main and PostProc). Lot of messages will appear on screen, and if everithing goes fine at the end you will see this message:

```
#################################################################
YOU HAVE DONE IT! YOU HAVE COMPILED THE MODEL!
LET'S START PLAYING WITH THE BEAST....
#################################################################
```

Congratulations! You can now go to next step and run a test simulation.

Chapter 4

Access global datasets

The first step to run a test simulation is to obtain static data to localize model DOMAIN and Atmosphere and Ocean global model dataset to build initial and boundary conditions ICBC to run a local area simulation.

ICTP maintains a public accessible web repository of datasets on:

http://users.ictp.it/ pubregcm/RegCM4/globedat.htm

As of now you are requested to download required global data on your local disk storage before any run attempt. In the future, the ICTP ESP team has planned to make available an OpenDAP THREDDS Server to give remote access to global dataset for creating DOMAIN and ICBC without the need to download the global dataset, but just the required subset in space and time, using the ICTP web server capabilities to create that subset.

4.1 Global dataset directory Layout

You are suggested to establish a convenient location for global datasets on your local storage. Keep in mind that required space for a year of global data can be as large as 8 GBytes.

Having this in mind, we will now consider that you the user have identified on your system or have network access to such a storage resource to store say 100 GB of data, and have it reachable on your system under the `$REGCM_GLOBEDAT` location. On this directory, you are required to make the following directories:

```
$> cd $REGCM_GLOBEDAT
$> mkdir SURFACE CLM SST AERGLOB EIN15
```

This does not fill all possible global data sources paths, but will be enough for the scope of running the model for testing its capabilities.

4.2 Static Surface Dataset

The model needs to be localized on a particular DOMAIN. The needed information are topography, land type classification and optionally lake depth (to run the Hostetler lake model) and soil texture classification (to run the chemistry option with DUST enabled).

This means downloading four files, which are global archives at 30*second* horizontal resolution on a global latitude-longitude grid of the above data.

```
$> cd $REGCM_GLOBEDAT
$> cd SURFACE
$> curl -o GTOPO_DEM_30s.nc.gz \
> http://clima-dods.ictp.it/data/d4/SURFACE/GTOPO_DEM_30s.nc.gz
$> gunzip GTOPO_DEM_30s.nc.gz
$> curl -o GLCC_BATS_30s.nc.gz \
> http://clima-dods.ictp.it/data/d4/SURFACE/GLCC_BATS_30s.nc.gz
$> gunzip GLCC_BATS_30s.nc.gz
```

Optional Lake and Texture datasets:

```
$> cd $REGCM_GLOBEDAT
$> cd SURFACE
$> curl -o ETOPO_BTM_30s.nc.gz \
> http://clima-dods.ictp.it/data/d4/SURFACE/ETOPO_BTM_30s.nc.gz
$> gunzip ETOPO_BTM_30s.nc.gz
$> curl -o GLZB_SOIL_30s.nc.gz \
> http://clima-dods.ictp.it/data/d4/SURFACE/GLZB_SOIL_30s.nc.gz
$> gunzip GLZB_SOIL_30s.nc.gz
```

4.3 Aerosol Database

If you are planning to enable aerosol in the model, you will need a single file, which contains sources of optical active species used in the model regridded from global model run.

```
$> cd $REGCM_GLOBEDAT
$> cd AERGLOB
$> curl -o AEROSOL.dat \
> http://clima-dods.ictp.it/data/d4/AEROSOL/AEROSOL.dat
```

This is the input file for the `aerosol` icbc program.

4.4 CLM Dataset

If you are planning to enable the `CLM` option in the model, you will need a series of files with global land surface characteristics datasets.

```
$> cd $REGCM_GLOBEDAT
$> cd CLM
$> CLMURL="clima-dods.ictp.it/data/d4/CLM"
$> curl -o mksrf_fmax.nc.gz \
> http://$CLMURL/mksrf_fmax.nc.gz
$> curl -o mksrf_glacier.nc.gz \
> http://$CLMURL/mksrf_glacier.nc.gz
$> curl -o mksrf_lai.nc.gz \
> http://$CLMURL/mksrf_lai.nc.gz
```

```
$> curl -o mksrf_lanwat.nc.gz \
> http://$CLMURL/mksrf_lanwat.nc.gz
$> curl -o mksrf_navyoro_20min.nc.gz \
> http://$CLMURL/mksrf_navyoro_20min.nc.gz
$> curl -o mksrf_pft.nc.gz \
> http://$CLMURL/mksrf_pft.nc.gz
$> curl -o mksrf_soicol_clm2.nc.gz \
> http://$CLMURL/mksrf_soicol_clm2.nc.gz
$> curl -o mksrf_soitex.10level.nc.gz \
> http://$CLMURL/mksrf_soitex.10level.nc.gz
$> curl -o mksrf_urban.nc.gz \
> http://$CLMURL/mksrf_urban.nc.gz
$> curl -o pft-physiology.c070207.gz \
> http://$CLMURL/pft-physiology.c070207.gz
$> curl -o pft-physiology.c070207.readme.gz \
> http://$CLMURL/pft-physiology.c070207.readme.gz
$> curl -o rdirc.05.061026.gz \
> http://$CLMURL/rdirc.05.061026.gz
$> gunzip *.gz
```

This is the input file for the `clm2rcm` program (see at 6.2.2).

4.5 Sea Surface Temperature

The model needs a global SST dataset to feed the model with ocean temperature. You have multiple choices for SST data:

1. *GISST* - UKMO SST (Rayner et al 1996), 1 degree from **http://www.badc.rl.ac.uk**
   UKMO DATA archive reformed as direct access binary format from the original ASCII format.
2. *OISST* - CAC Monthly Optimal Interpolation dataset in the original netCDF format.
3. *OI2ST* - Same as above, but both SST and Sea Ice dataset (used if seaice option is enabled in the model).
4. *OI_WK* - OISST CAC Weekly Optimal Interpolation dataset in the original netCDF format
5. *OI2WK* - Same as above, but both SST and Sea Ice dataset
6. *EH5RF* - EC-MPI 6 hourly 1.875x1.875, reference from 1941 to 2000
7. *EH5A2* - Same as above, from 2001 to 2100 IPCC A2 scenario
8. *EH5B1* - Same as above, from 2001 to 2100 IPCC B1 scenario
9. *EHA1B* - Same as above, from 2001 to 2100 IPCC A1B scenario
10. *ERSST* - ERA interim Project 6 hourly 1.5x1.5 degree SST

11. *ERSKT* - ERA interim as above but Skin temperature
12. *FV_RF* - HadAMH_SST in the original netCDF format, from 1959 to 1991
13. *FV_A2* - Same as above, IPCC A2 scenario
14. *FV_B2* - Same as above, IPCC B2 scenario
15. *CCSST* - CCSM3 POP gx1v3 regridded 1x1 data

We will for now for our test run download just CAC OISST weekly for the period 1981 - present.

```
$> cd $REGCM_GLOBEDAT
$> cd SST
$> CDCSITE="ftp.cdc.noaa.gov/pub/Datasets/noaa.oisst.v2"
$> curl -o sst.wkmean.1981-1989.nc \
> ftp://$CDCSITE/sst.wkmean.1981-1989.nc
$> curl -o sst.wkmean.1990-present.nc \
> ftp://$CDCSITE/sst.wkmean.1990-present.nc
```

4.6 Atmosphere and Land temperature Global Dataset

The model needs to build initial and boundary conditions for the regional scale, interpolating on the RegCM grid the data from a Global Climatic Model output. The GCM dataset can come from any of the supported models:

1. *EINXX* - ECMWF INTERIM 10 year reanalysis datasets, XX can have values 25, 15 or 75 for resolution 2.5x2.5L37, 1.5x1.5L37, 0.75x0.75L37. Time resolution is 4 times daily.
2. *ECMWF* - ECMWF TOGA/WCRP Uninitialized Data - (ECWCRP). Reformatted by PWC/ICTP to direct-access binary, T42L15, Gaussian Grid. [1]
3. *ERA40* - ECMWF 40 year reanalysis datasets, available from **http://data.ecmwf.int/data/d/era40_daily**, Pressure levels, 2.5x2.5L23, 4 times daily.
4. *ERAHI* - ECMWF 40 year reanalysis datasets, origigal model level fields: T, U, V and log(Ps) are in spectral coefficients, Oro and Q are at the reduced Gaussian grids. T159L60 (N80L60).
5. *NNRPY* - NCEP/NCAR Reanalysis datasets, Y can have values 1 and 2 for the two reanalysis experimens. Data are available in the original netCDF format at **ftp://ftp.cdc.noaa.gov/Datasets/ncep.reanalysis** (1948 – *present*, 2.5x2.5L13) and **ftp://ftp.cdc.noaa.gov/Datasets/ncep.reanalysis2** (1979 – 2009, 2.5x2.5L13).

[1] As of RegCM 4.1 this input source is not tested

6. *NRP2W* - Small Window (instead of global) of NNRP1/2 to save disk space. This window can be created from original files with NCO tools. [2]

7. *GFS11* - NCEP Global Forecast System (GFS) product FNL, from **http://dss.ucar.edu/datasets/ds083.2/data/fnl-yyyymm**, Pressure levels, 1.0x1.0L27, 4 times daily.

8. *FVGCM* - FVGCM run by the PWC group of Abdus Salam ICTP. [3]

9. *EH5OM* - EH5OM run by the MPI at Hamburg, T63, Gaussian grid. For present day run: $1941 - 2000$, for A1B scenario run: $2001 - 2100$. 17 pressure levels, 4 times daily, direct-access binary.

10. *CCSMN* - unpacked CCSM3 NETCDF L26 (six hourly) data, either global or window, can be obtained from **http://www.earthsystemgrid.org**

11. *FNEST* - Further oneway NESTing from previous RegCM run.

We will for now for our test run download just the EIN15 dataset for the year 1990 (Jan 01 00:00:00 UTC to Dec 31 18:00:00 UTC)

```
$> cd $REGCM_GLOBEDAT
$> cd EIN15
$> mkdir 1990
$> cd 1990
$> ICTPSITE="clima-dods.ictp.it/data/d9/ERAIN150/1990/"
$> for type in "air hgt rhum uwnd vwnd"
>  do
>    for hh in "00 06 12 18"
>    do
>      curl -o ${type}.1990.${hh}.nc \
>         http://$ICTPSITE}/${type}.1990.${hh}.nc
>    done
>  done
```

With this dataset we are now ready to go through the RegCM Little Tutorial in the next chapter of this User Guide.

[2] As of RegCM 4.1 this input source is not tested
[3] As of RegCM 4.1 this input source is not tested

Chapter 5

Run a test simulation using the model

We will in this chapter go through a sample session in using the model with a sample configuration file prepared for this task.

5.1 Setting up the run environment

The model executables prepared in chapter 3 are waiting for us to use them. So let's give them a chance.

The model test run proposed here requires around 100Mb of disk space to store the DOMAIN and ICBC in input and the output files. We will assume here that you, the user, have already established a convenient directory on a disk partition with enough space identified in the following discussion with `$REGCM_RUN`

We will setup in this directory a standard environment where the model can be executed for the purpose of learning how to use it.

```
$> cd $REGCM_RUN
$> mkdir input output
$> ln -sf $REGCM_ROOT/Bin .
$> cp $REGCM_ROOT/Testing/test_001.in .
$> cd $REGCM_RUN
```

Now we are ready to modify the input namelist file to reflect this directory layout. A namelist file in Fortran90 is a convenient way to give input to a program in a formatted file, read at runtime by the program to setup its execution behaviour. So the next step is somewhat tricky, as you need to edit the namelist file respecting its well defined syntax. Open your preferred text file editor and load the `test_001.in` file. You will need to modify for the scope of the present tutorial the following lines:

```
FROM:
 dirter = '/set/this/to/where/your/domain/file/is',
TO:
 dirter = 'input/',
```

```
FROM:
 inpter = '/set/this/to/where/your/surface/dataset/is',
TO:
 inpter = '$REGCM_GLOBEDAT',
```

where `$REGCM_GLOBEDAT` is the directory where input data have been downloaded in chapter 4.

```
FROM:
 dirglob = '/set/this/to/where/your/icbc/for/model/is',
TO:
 dirglog = 'input/',

FROM:
 inpglob = '/set/this/to/where/your/input/global/data/is',
TO:
 inpglob = '$REGCM_GLOBEDAT',
```

and last bits:

```
FROM:
 dirout='/set/this/to/where/your/output/files/will/be/written'
TO:
 dirout='output/'
```

These modifications just reflect the above directory layout proposed for this tutorial, and any of these paths can point anywhere on your system disks. The path is limited to 256 characters. We are now ready to execute the first program of the RegCM model.

5.2 Create the DOMAIN file using terrain

The first step is to create the DOMAIN file to localize the model on a world region. The program which does this for you reading the global databases is `terrain` .

To launch the terrain program, enter the following commands:

```
$> cd $REGCM_RUN
$> ./Bin/terrain test_001.in
```

If everything is correctly configured up to this point, the model should print something on stdout, and the last lines will be:

```
 Grid data written to output file
 Successfully completed terrain fields generation
```

In the input directory the program will write the following two files:

```
$> ls input
test_001_DOMAIN000.nc test_001_LANDUSE
```

The DOMAIN file contains the localized topography and landuse databases, as well as projection information and land sea mask. The second file is an ASCII encoded version of the landuse, used for modifying it on request. We will cover it's usage later on. To have a quick look at the DOMAIN file content, you may want to use the GrADSNcPlot program:

```
$> ./Bin/GrADSNcPlot input/test_001_DOMAIN000.nc
```

If not familiar with GrADS program, enter in sequence the following commands at the `ga->` prompt:

```
ga-> q file
ga-> set gxout shaded
ga-> set mpdset hires
ga-> set cint 50
ga-> d topo
ga-> c
ga-> set cint 1
ga-> d landuse
ga-> quit
```

this will plot the topography and the landuse on the X11 window.

5.3 Create the SST using the sst program

We are now ready to create the Sea Surface Temperature for the model, reading a global dataset. The program which does this for you is the `sst` program, which is executed with the following commands:

```
$> cd $REGCM_RUN
$> ./Bin/sst test_001.in
```

If everything is correctly configured up to this point, the model should print something on stdout, and the last line will be:

```
 Successfully generated SST
```

The input directory now contains a new file:

```
$> ls input
test_001_DOMAIN000.nc test_001_LANDUSE test_001_SST.nc
```

The SST file contains the Sea Surface temperature to be used in generating the Initial and Boundary Conditions for the model for the period specified in the namelist file. Again you may want to use the GrADSNcPlot program to look at file content:

```
$> ./Bin/GrADSNcPlot input/test_001_SST.nc
```

If not familiar with GrADS program, enter in sequence the following commands at the `ga->` prompt:

```
ga-> q file
ga-> set gxout shaded
ga-> set mpdset hires
ga-> set cint 2
ga-> d sst
ga-> quit
```

this will plot the interpolated sst field on the X11 window.

5.4 Create the ICBC files using the icbc program

Next step is to create the ICBC (Initial Condition, Boundary Conditions) for the model itself. The program which does this for you is the `icbc` program, executed with the following commands:

```
$> cd $REGCM_RUN
$> ./Bin/icbc test_001.in
```

If everything is correctly configured up to this point, the model should print something on stdout, and the last line will be:

```
 Successfully completed ICBC
```

The input directory now contains two more files:

```
$> ls -1 input
test_001_DOMAIN000.nc
test_001_ICBC.1990060100.nc
test_001_ICBC.1990070100.nc
test_001_LANDUSE
test_001_SST.nc
```

The ICBC files contain the surface pressure, surface temperature, horizontal 3D wind components, 3D temperature and mixing ratio for the RegCM domain for the period and time resolution specified in the input file. Again you may want to use the GrADSNcPlot program to look at file content:

```
$> ./Bin/GrADSNcPlot input/test_001_ICBC.1990060100.nc
```

If not familiar with GrADS program, enter in sequence the following commands at the `ga->` prompt:

```
ga-> q file
ga-> set gxout shaded
ga-> set mpdset hires
ga-> set cint 2
ga-> d ts
ga-> c
ga-> set lon 10
ga-> set lat 43
```

```
ga-> set t 1 last
ga-> d ts
ga-> quit
```

this will plot the interpolated surface temperature field on the X11 window, first at first time step and then a time section in one of the domain points for a whole month.

We are now ready to run the model!

5.5 First RegCM model simulation

The model has now all needed data to allow you to launch a test simulation, the final goal of our little tutorial.

The model command line now will differ if you have prepared the Serial or the MPI version. For the MPI enabled version we will assume that your machine is a dual core processor (baseline for current machines, even for laptops). Change the `-np 2` argument to the number of processors you have on Your platform (on my laptop QuadCore I use `-np 4`).

- MPI version

  ```
  $> cd $REGCM_RUN
  $> mpirun -np 2 ./Bin/regcmMPI test_001.in
  ```

- Serial version [1]

  ```
  $> cd $REGCM_RUN
  $> ./Bin/regcmSerial test_001.in
  ```

Now the model will start running, and a series of diagnostic messages will be printed on screen. As this is a simulation known to behave well, no stoppers will appear, so you may want now to have a coffee break and come back in 10 minutes from now.

At the end of the run, the model will print the following message:

```
 RegCM V4 simulation successfully reached end
```

The output directory now contains four files:

```
$> ls output
test_001_ATM.1990060100.nc test_001_SRF.1990060100.nc
test_001_RAD.1990060100.nc test_001_SAV.1990070100
```

the ATM file contains the atmosphere status from the model, the SRF file contains the surface diagnostic variables, and the RAD file contains radiation fluxes information. The SAV file stores the status of the model at the end of the simulation period to enable a restart, thus allowing a long simulation period to be splitted in shorter simulations.

To have a look for example at surface fields, you may want to use the following command:

[1] Deprecated. Support will be dropped in future releases.

```
$> ./Bin/GrADSNcPlot output/test_001_SRF.1990060100.nc
```

Assuming the previous crash course in using GraDS was received, you should be able to plot the variables in the file.

This is the end of this little tutorial, and in the next chapter we will examine how to configure the model for your research needs.

Chapter 6

Localize the model and run your simulation

We will examine in this chapter in more detail the namelist configuration file, to give you the User a deeper knowledge of model capabilities.

6.1 The commented namelist

In this section we will show you the commented namelist input file you will find under `$REGCM_ROOT/Doc` with the name `README.namelist` . All model programs seen so far, with the exception of the GrADS helper program, use as input this namelist file, which is unique to a particular simulation. The model input namelist file is divided in stanzas, each one devoted to configuring the model capabilities. A stanza in the namelist is identified with a starting `&` character followed by stanza name, and ends on a single line with the \ character.

6.1.1 dimparam stanza

This stanza contains the base X,Y,Z domain dimension information, used by the model dynamic memory allocator to request the Operating System the memory space to store the model internal variables.

```
&dimparam
iy     = 34,  ! This is number of points in the N/S direction
jx     = 48,  ! This is number of points in the E/W direction
kz     = 18,  ! Number of vertical levels: supported are 14, 18 and 23
nsg    = 1,   ! For subgridding, number of points to decompose. If nsg=1,
              ! no subgridding is performed. CLM does NOT work as of now with
              ! subgridding enabled.
/
```

The things you need to know here:

1. In the current version 4.1 the model parallelizes execution dividing the work between the processors along the jx (longitude) dimension. The minimum work per processor is 3 points along the jx dimension, so the maximum number of processors which can be used in a parallel run for the above configuration is just 16. In future revision ICTP plans to introduce 2D decomposition.

2. In the future model revision the kz (vertical) dimension will be more configurable, but for now you are limited to have 14, 18 and 23 levels. We at ICTP normally use 18 levels.

3. Specifying an nsg number greater than one triggers the subgrid BATS model on. There is no plan to extend this feature to CLM model. This affects only surface variable calculations. All dynamical variables are calculated still on the coarser grid. Rain in the current implementation is also calculated on the coarser grid.

6.1.2 geoparam stanza

This stanza is used by the `terrain` program to geolocate the model grid on the earth surface. The RegCM model uses a limited number of projection engines. The value here are used by the other model programs to assert consistency with the geolocation information written by the `terrain` program in the `DOMAIN` file.

The first step in any application is the selection of model domain and resolution. There are no strict rules for this selection, which in fact is mostly determined by the nature of the problem and the availability of computing resources. The domain should be large enough to allow the model to develop its own circulations and to include all relevant forcings and processes, and the resolution should be high enough to capture local processes of interest (e.g. due to complex topography or land surface).

On the other hand the model computational cost increases rapidly with resolution and domain size, so a compromise needs to be usually reached between all these factors.

This is usually achieved by experience, understanding of the problem or trial and error, however one tip to remember is to avoid that the boundaries of the domain cross major topographical systems.

This is because the mismatch in the resolution of the coarse scale lateral driving fields and the model fields in the presence of steep topography may generate spurious local effects (e.g. localized precipitation areas) which can affect the model behavior, at least in adjacent areas.

```
&geoparam
iproj = 'LAMCON', ! Domain cartographic projection. Supported values are:
                  ! 'LAMCON', Lambert conformal.
                  ! 'POLSTR', Polar stereographic. (Doesn't work)
                  ! 'NORMER', Normal  Mercator.
                  ! 'ROTMER', Rotated Mercator.
ds = 60.0,        ! Grid point horizontal resolution in km
ptop = 5.0,       ! Pressure of model top in cbar
clat = 45.39,     ! Central latitude  of model domain in degrees
                  ! North hemisphere is positive
clon = 13.48,     ! Central longitude of model domain in degrees
                  ! West is negative.
plat = 45.39,     ! Pole latitude (only for rotated Mercator Proj)
plon = 13.48,     ! Pole longitude (only for rotated Mercator Proj)
truelatl = 30.0,  ! Lambert true latitude (low latitude side)
truelath = 60,    ! Lambert true latitude (high latitude side)
i_band = 0,       ! Enable ONLY if BAND option activated.
/
```

The things you need to know here:

1. The different projection engines produce better results depending on the position and extent of the domain. In particular, regardless of hemisphere:
   - Middle latitudes (around 45 degrees) - Lambert Conformal
   - Polar latitudes (more than 75 degrees) - Polar Stereographic
   - Low latitudes (up to 30 degrees and crossing the equator) - Mercator
   - Crossing more than 45 degrees extent in latitude - Rotated Mercator
2. The model hydrostatic engine does not allow a resolution lower than $20km$. If you want a higher resolution consider using the subgridding scheme. ICTP plans to introduce in the future a non-hydrostatic compressible core to the RegCM model.
3. Lowering the top pressure of the model can give you problems in regions with complex topography. Touch the default after thinking twice on that.
4. Always specify `clat` and `clon`, the central domain point, and do fine adjustment of the position moving it around a little bit. A little shift in position and some tests can help you obtain a better representation of coastlines and topography at the coarse resolutions.
5. If using `LAMCON` projection, take care to place the two true latitudes at around one fourth and three fourth of the domain latitude space to better correct the projection distortion of the domain.
6. The pole position for the rotated mercator position should be as near as possible to the center domain position.
7. For the `i_band` parameter, see below in the `BAND` option discussion in 6.2.1.

6.1.3 aerosolparam stanza

This stanza allows the user to specify aerosol usage in the model. It does enable building of soil texture database in the `terrain` program and controls the dimension of the number of optical active tracers used in the active chemistry tracers part of the model.

```
&aerosolparam
aertyp = 'AER00D0' ! Aerosol dataset used
                   ! One in :
                   ! AER00D0 -> Neither aerosol, nor dust used
                   ! AER01D0 -> Biomass, SO2 + BC + OC, no dust
                   ! AER10D0 -> Anthropogenic, SO2 + BC + OC, no dust
                   ! AER11D0 -> Anthropogenic+Biomass, SO2 + BC + OC, no dust
                   ! AER00D1 -> No aerosol, with dust
                   ! AER01D1 -> Biomass, SO2 + BC + OC, with dust
                   ! AER10D1 -> Anthropogenic, SO2 + BC + OC, with dust
                   ! AER11D1 -> Anthropogenic+Biomass, SO2 + BC + OC, with dust
ntr =  4,          ! Tracer parameters: number of tracers
nbin = 2,          ! Tracer parameters: bins number for dust
/
```

The things you need to know here:

1. If `aertyp` is left to `AER00D0`, it is nonsense to activate chemistry in the model.

2. The total number of tracers activated must be greater than nbin.

3. If Anthropogenic and/or Biomass is activated, the model will also need the user to run the aerosol program. It can be run at the same level as the sst program, with the same calling syntax. Just replace sst with aerosol.

   ```
   $> cd $REGCM_RUN
   $> ./Bin/aerosol myregcm.in
   ```

   The aerosol program prepares an emission dataset used by the model to consider optical active chemistry species effects in the radiation calculation. The surface dust emission are calculated using the soil texture dataset prepared by the `terrain` program if the last 0 is set to 1 in `aertyp`.

6.1.4 terrainparam stanza

This stanza is used by the `terrain` program to know how you want to generate the `DOMAIN` file. You can control its work using a number of parameters to obtain what you consider the best representation of the physical reality. Do not underestimate what you can do at this early stage, having a good representation of the surface can lead to valuable results later when the model calculates climatic parameters.

```
&terrainparam
domname  = 'AQWA',      ! Name of the domain. Controls naming of input files
ntypec = 5,             ! Resolution of the global terrain and landuse data
                        ! Use 60, for  1  degree resolution
                        !     30, for 30 minutes resolution
                        !     10, for 10 minutes resolution
                        !      5, for  5 minutes resolution
                        !      3, for  3 minutes resolution
                        !      2, for  2 minutes resolution
ntypec_s = 2,           ! Same for subgrid (Used only if nsg > 1)
smthbdy = .false.,      ! Smoothing Control flag
                        !  true  -> Perform extra smoothing in boundaries
lakedpth    = .false.,  ! If using lakemod (see below), produce from
                        ! terrain program the domain bathymetry
fudge_lnd   = .false.,  ! Fudging Control flag, for landuse of grid
fudge_lnd_s = .false.,  ! Fudging Control flag, for landuse of subgrid
fudge_tex   = .false.,  ! Fudging Control flag, for texture of grid
fudge_tex_s = .false.,  ! Fudging Control flag, for texture of subgrid
fudge_lak   = .false.,  ! Fudging Control flag, for lake of grid
fudge_lak_s = .false.,  ! Fudging Control flag, for lake of subgrid
h2opct = 75.,           ! Surface minimum H2O percent to be considered water
dirter = 'input/',      ! Output directory for terrain files
inpter = 'globdata/',   ! Input directory for SURFACE dataset
/
```

The things you need to know here:

1. The `domname` will control the output file naming convention, all generated files will add this prefix to the old V3 naming convention, giving you the capability to recognize different runs. Try to use always meaningful names.

2. In version 4.1 does exist a single input dataset, the 30s one. The `ntypec` parameter controls initial subsampling of input dataset before the smoothing interpolation performed by the `terrain` program.

3. Use lakedepth if you plan to use the Hostetler lake model later on. It will be useless otherwise. It may be used in the future to have a common sea bathymetry with the ocean coupled model. The coupling engine of RegCM will be included in future model releases.

4. You can control the final land-water mask using the h2opct parameter. This parameter can be used to have more land points than calculated by the simple interpolation engine. Try it with different values to find best land shapes. A zero value means use just the interpolation engine, higher values will extend into ocean points the land at land-water interface.

5. A number of flags control the capability of the `terrain` program to modify on request the class type variables in the `DOMAIN` file. You can modify on request the landuse, the texture and the lake/land interface. Running once the `terrain` program, it will generate for you aside from the `DOMAIN` file a series of ASCII files you can modify with any text editor. Running the `terrain` program the second time and setting a `fudge` flag, will tell the program to overwrite the selected variable with the modified value in the ASCII file. This can be useful for sensitivity experiments in the BATS surface model or to design a scenario experiment.

6. Some of the land surface types in BATS have been little tested and used or are extremely simplified and thus should be used cautiously. Specifically the types are: sea ice, bog/marsh, irrigated crop, glacier. If such types are present in a domain, the user is advised to carefully check the model behavior at such points and eventually substitute these types with others.

7. The `inpter` directory is expected to contain a `SURFACE` directory where the actual netCDF global dataset are stored. The overall path is limited to 256 characters.

8. If the netCDF library is compiled with OpenDAP support, an URL can be used as a path in the `dirter` and `inpter` variables. Note that the 256 character limit for paths holds in the whole program. For `terrain` program you may want to try the following URL:
http://clima-dods.ictp.it/thredds/dodsC

9. The texture dataset is built if the aerosol model is activated. This is controlled by the `AERTYP` flag. See above in 6.1.3.

6.1.5 globdatparam stanza

This stanza is used by the `sst` and `icbc` ICBC programs. You can tell them how to build initial and bondary conditions.

```
&globdatparam
ibdyfrq =     6,            ! boundary condition interval (hours)
ssttyp = 'OI_WK',           ! Type of Sea Surface Temperature used
                            !  One in: GISST, OISST, OI2ST, OI_WK, OI2WK,
                            !          FV_RF, FV_A2, FV_B2,
```

```
                            !          EH5RF, EH5A2, EH5B1, EHA1B,
                            !          ERSST, ERSKT, CCSST
dattyp = 'EIN15',           ! Type of global analysis datasets used
                            !  One in: ECMWF, ERA40, EIN75, EIN15, EIN25,
                            !          ERAHI, NNRP1, NNRP2, NRP2W, GFS11,
                            !          FVGCM, FNEST, EH5OM , CCSMN
ehso4 = .false.,            ! Sulfate dataset Control Flag
globidate1 = 1990060100,    ! Start date for ICBC data generation
globidate2 = 1990070100,    ! End data for ICBC data generation
dayspy  = 365.2422D0,       ! Number of days per year
dirglob = 'input/',         ! Path for ICBC produced input files
inpglob = 'globdata/',      ! Path for ICBC global input datasets.
                  ! Look http://users.ictp.it/~pubregcm/RegCM4/globedat.htm
                  ! on how to download them.
/
```

Things you need to know here:

1. The globidate time window to build ICBC must be always greater or equal to the time window you plan to run the model in. Different GCMs and reanalysis products have different length of the year. For example, the reanalysis products employ the real year length (365 days + real leap years, i.e. and average length of 365.2422), the CCSM has a length of 365 days (no leap year), the HadCM has a length of 360 days (30 day months). The RegCM4 length of the year has to be the same as in the forcing fields, and this can be set in the variable `dayspy`. Please remember to always check the consistency of the length of the year.

2. Even if listed, not all the input engines are fully tested. Some of them need data which have been reformatted by ICTP (they are not in the original format with which they are distributed by the institution producing them). Some input data are not freely distibutable by ICTP, and you need a special agreement with the owner to use them. Hopefully the situation is changing, and data exchange is becoming more and more the basis for good science in the climatic field.

3. For notes on path, you can see the above in terrainparam stanza description at 7.

6.1.6 ioparam stanza

```
&ioparam
ibyte = 4,   ! Number of bytes in reclen. Usually 4
/
```

Leave this untouched. The model expects input record syze to be 4 bytes. You will need to change some compilation parameters if you change this value. [1]

6.1.7 debugparam stanza

This stanza is used by all RegCM programs to enable/disable some debug printout. In the current release this flag is honored only by the model itself. If you are not a developer you may find this flags useless.

[1] This namelist stanza will be removed in future versions

```
&debugparam
debug_level = 1, ! Currently value of 2 and 3 control previous DIAG flag
dbgfrq = 3,      ! Interval for printout if debug_level >= 3
/
```

Just note that with current implementation, the output file syncing is left to the netCDF library. If You want to examine step by step the output while the model is running, set the `debug_level` at value 3.

6.1.8 boundaryparam stanza

Being a limited area model, in order to be run RegCM4 requires the provision of meteorological initial and time dependent lateral boundary conditions, typically for wind components, temperature, water vapor and surface pressure. These are obtained by interpolation from output from reanalysis of observations or global climate model simulations, which thus drive the regional climate model.

The lateral boundary conditions (LBC) are provided through the so called relaxation/diffusion technique which consists of:

1. selecting a lateral buffer zone of n grid point width (`nspgx`)
2. interpolating the driving large scale fields onto the model grid
3. applying the relaxation + diffusion term

$$\frac{\partial \alpha}{\partial t} = F(n)F_1 * (\alpha_{LBC} - \alpha_{mod}) - F(n)F2 * \Delta_2(\alpha_{LBC} - \alpha_{mod}) \quad (6.1)$$

where α is a prognostic variable (wind components, temperature, water vapor, surface pressure). The first term on the rhs is a Newtonian relaxation term which brings the model solution (*mod*) towards the LBC field (*LBC*) and the second term diffuses the differences between model solution and LBC. $F(n)$ is an exponential function given by:

$$F(n) = exp\left(\frac{-(n-1)}{anudge(k)}\right) \quad (6.2)$$

Where n is the grid point distance from the boundary (varying from 1 to $nspgx$): $n-1$ is the outermost grid point, $n = 2$ the adjacent one etc. The *anudge* array determines the strength of the LBC forcing and depends on the model level k. In practice $F(n)$ is equal to 1 at the outermost grid point row and decreases exponentially to 0 at the internal edge of the buffer zone ($nspgd$) at a rate determined by *anudge*. Larger buffer zones and larger values of *anudge* will yield a greater forcing by the LBC.

Typically for domain sizes of 100 grid points we use a buffer zone width of $10-12$ grid points, for large domains this buffer zone can increase to values of 15 or even 20.

In the model *anudge* has three increasing values from the lower, to the mid and higher troposphere. For example for $nspgx = 10$ we use *anudge* equals to $1, 2, 3$ for the lower, mid and upper troposphere, respectively.

This allows a stronger forcing in the upper troposphere to insure a greater consistency of large scale circulations with the forcing LBC while allowing more

freedom to the model in the lower troposphere where local high resolution forcings (e.g. complex topography) are more important.

For nspgx of $15-20$, for example, *anudge* values could be increased to $2, 3, 4$. As a rule of thumb, the choice of the maximum *anudge* value should follow the conditions:

$$\frac{(nspgx-1)}{anudge(k)} \geq 3 \tag{6.3}$$

```
&boundaryparam
nspgx  = 12, ! nspgx-1 represent the number of cross point slices on
             ! the boundary sponge or relaxation boundary conditions.
nspgd  = 12, ! nspgd-1 represent the number of dot point slices on
             ! the boundary sponge or relaxation boundary conditions.
high_nudge   =     3.0, ! Nudge value high range
medium_nudge =     2.0, ! Nudge value medium range
low_nudge    =     1.0  ! Nudge value low range
/
```

6.1.9 modesparam stanza

This needs not to be changed. Leave it to the default value.

```
&modesparam
nsplit = 2, ! Number od split exp modes
/
```

6.1.10 restartparam stanza

This stanza lets you control the time period the model is currently simulating in this particular run. You may want to split longer runs for which you have prepared the ICBC's into shorter runs, to schedule HPC resource usage in a more collaborative way with other researcher sharing it: the regcm model allows restart, so be friendly with other research projects which may not have this fortune (unless you are late for publication).

```
&restartparam
ifrest  = .false. ,   ! If a restart
idate0  = 1990060100, ! Global start (is globidate1)
idate1  = 1990060100, ! Start date of this run
idate2  = 1990060200, ! End date for this run
/
```

Things you need to know here:

1. After the simulation starts, on restart NEVER change the `idate0` value. The correct scheme for restart is:
   - Set `ifrest` to `.true.`
   - Set `idate1` to the value in `idate2`
   - Define the new value for `idate2`
2. Consider that current RegCM convention is to place midnight of first day of month as the last timestep in previous month, except on first model output file (`ifrest` = `.false.`). It is for this reason better to use as start and end time a month boundary. We usually consider a month data file the basic unit of output, each time you cross a month a new output file will be created for you.

6.1.11 timeparam stanza

This stanza contains model internal timesteps, used by the model as basic integration timestep and triggers for calling internal parametric schemes.

```
&timeparam
dt      =   150.,   ! time step in seconds
radfrq  =    30.,   ! time interval in min solar rad caluclated
abemh   =    18.,   ! time interval absorption-emission calculated (hours)
abatm   =   600.,   ! time interval at which bats is called (secs)
/
```

Things you need to know here:

1. The dynamical hydrostatical core of RegCM requires a fixed timestep, and you need to manually find the correct value which permits not to break the CourantFriedrichsLewy condition considering *R. Courant and Lewy* (1928). A good rule of thumb is to have a `dt` not greater than three times the `ds` value in *km* specified in the `geoparam` stanza at 6.1.2. A greater value may lower computing time, but in case of strong advection may lead to non accurate computation or even the violation of CFL condition and the divergence of the solution.

2. All the other internal timesteps need to be multiples of the base timestep. Note that the units are different, so you need to convert the other timesteps in seconds before the check.

3. In case of strong surface gradients, a low value for the surface timesteps may help the model better describe the interaction with the atmosphere and obtain a stable solution.

4. If you hit a non stable condition, the restart capability of the model may help find the correct timestep just for a particular period, using a different timestep at different times.

6.1.12 outparam stanza

This stanza controls the model output engine, allowing you to enable/disable any of the output file writeout, or to modify the frequency the fields are written in the files.

```
&outparam
ifsave  = .true. ,    ! Create SAV files for restart
savfrq  =    48.,     ! Frequency in hours to create them
iftape  = .true. ,    ! Output ATM ?
tapfrq  =     6.,     ! Frequency in hours to write to ATM
ifrad   = .true. ,    ! Output RAD ?
radisp  =     6.,     ! Frequency in hours to write to RAD
ifbat   = .true. ,    ! Output SRF ?
ifsub   = .true. ,    ! Output SUB ?
batfrq  =     3.,     ! Frequency in hours to write to SRF and SUB (and CLM)
iflak   = .true.,     ! Output LAK ?
lakfrq  =     6.,     ! Frequency in hours to write to LAK if lakemod is 1
                      ! It must be an integer multiple of batfrq
ifchem  = .true.,     ! Output CHE ?
chemfrq =     6.,     ! Frequency in hours to write to CHE
dirout  = './output', ! Path where all output will be placed
/
```

Things you need to know here:

1. The surface fields are the mean values in the interval specified by the frequency values. The dynamical fields are instead the point value at the output time. Refer to the Reference Manual *Giorgi* (2011) for a detailed description of the model output fields.

2. If the chemistry or lake model are not enabled, the values specified in the control flags are not considered. If `nsg` is not greater than one in dimparam at 6.1.1, the `ifsub` flag is not considered.

3. For the output directory, the path variable has a limit of 256 characters. This path must be a local path on disk where the user running the model has write permissions granted.

6.1.13 physicsparam stanza

This stanza controls the model physics. You have a number of option here, and the best way to select the right set is to carefully read the the Reference Manual *Giorgi* (2011). We are for the purposes of this User Guide not going in detail in here, except in saying that probably you will need to run some experiments especially with different cumulus convection schemes before finding out the best model setting. Although the mixed convection scheme (Grell over land and Emanuel over ocean) seems to provide an overall better performance, our experience is that there is no scheme that works best everywhere, therefore we advice to always do some sensitivity experiments to select the best scheme for your application.

```
&physicsparam
iboudy  =          5,  ! Lateral Boundary conditions scheme
                       !   0 => Fixed
                       !   1 => Relaxation, linear technique.
                       !   2 => Time-dependent
                       !   3 => Time and inflow/outflow dependent.
                       !   4 => Sponge (Perkey & Kreitzberg, MWR 1976)
                       !   5 => Relaxation, exponential technique.
ibltyp  =          1,  ! Boundary layer scheme
                       !   0 => Frictionless
                       !   1 => Holtslag PBL (Holtslag, 1990)
icup    =          4,  ! Cumulus convection scheme
                       !   1 => Kuo
                       !   2 => Grell
                       !   3 => Betts-Miller (1986) DOES NOT WORK !!!
                       !   4 => Emanuel (1991)
                       !  99 => Use Grell over land and Emanuel over ocean
                       !  98 => Use Emanuel over land and Grell over ocean
  igcc  =          1,  ! Grell Scheme Cumulus closure scheme
                       !   1 => Arakawa & Schubert (1974)
                       !   2 => Fritsch & Chappell (1980)
ipptls  =          1,  ! Moisture scheme
                       !   1 => Explicit moisture (SUBEX; Pal et al 2000)
iocnflx =          2,  ! Ocean Flux scheme
                       !   1 => Use BATS1e Monin-Obukhov
                       !   2 => Zeng et al (1998)
  iocnrough =      1,  ! Zeng Ocean model roughness formula to use.
                       !   1 => (0.0065*ustar*ustar)/egrav
                       !   2 => (0.013*ustar*ustar)/egrav + 0.11*visa/ustar
ipgf    =          0,  ! Pressure gradient force scheme
```

```
                      !   0 => Use full fields
                      !   1 => Hydrostatic deduction with pert. temperature
iemiss  =          0,  ! Calculate emission
lakemod =          0,  ! Use lake model
ichem   =          1,  ! Use active aerosol chemical model
scenario =      'A1B', ! IPCC Scenario to use in A1B,RF,A2,B1,B2
idcsst   =         0, ! Use diurnal cycle sst scheme
iseaice  =         0, ! Model seaice effects
idesseas =         1, ! Model desert seasonal albedo variability
iconvlwp =         1, ! Use convective liquid water path as the large-scale
                      ! liquid water path
\
```

6.1.14 subexparam stanza

This stanza controls the moisture scheme. Please consider carefully reporting in your work the tuning you perform on this parameters. The parameters below are the ones currently used at ICTP.

```
&subexparam
ncld      =          1, ! # of bottom model levels with no clouds
fcmax     =       0.80, ! Maximum cloud fraction cover
qck1land  =   .250E-03, ! Autoconversion Rate for Land
qck1oce   =   .250E-03, ! Autoconversion Rate for Ocean
gulland   =        0.4, ! Fract of Gultepe eqn (qcth) when precip occurs
guloce    =        0.4, ! Fract of Gultepe eqn (qcth) for ocean
rhmax     =       1.01, ! RH at whicn FCC = 1.0
rh0oce    =       0.90, ! Relative humidity threshold for ocean
rh0land   =       0.80, ! Relative humidity threshold for land
tc0       =      238.0, ! Below this temperature, rh0 begins to approach unity
cevap     =   .100E-02, ! Raindrop evap rate coef [[(kg m-2 s-1)-1/2]/s]
caccr     =      3.000, ! Raindrop accretion rate [m3/kg/s]
cllwcv    =     0.3E-3, ! Cloud liquid water content for convective precip.
clfrcvmax =       0.25, ! Max cloud fractional cover for convective precip.
cftotmax  =       0.75, ! Max total cover cloud fraction for radiation
/
```

We found that RegCM4 is especially sensitive to:

1. `cevap` : increasing cevap will generally decrease precipitation
2. `gulland`, `guloce` : increase of guland/guloce will generally lead to reduce precipitation

6.1.15 grellparam and emanparam stanzas

You are allowed here to tune the convection scheme selected above in 6.1.13 with the `icup` number if selected number is $2, 4, 98, 99$.

```
&grellparam
shrmin = 0.25,       ! Minimum Shear effect on precip eff.
shrmax = 0.50,       ! Maximum Shear effect on precip eff.
edtmin = 0.25,       ! Minimum Precipitation Efficiency
edtmax = 0.50,       ! Maximum Precipitation Efficiency
edtmino = 0.25,      ! Minimum Precipitation Efficiency (o var)
edtmaxo = 0.50,      ! Maximum Precipitation Efficiency (o var)
edtminx = 0.25,      ! Minimum Precipitation Efficiency (x var)
edtmaxx = 0.50,      ! Maximum Precipitation Efficiency (x var)
shrmin_ocn = 0.25,   ! Minimum Shear effect on precip eff. OCEAN points
shrmax_ocn = 0.50,   ! Maximum Shear effect on precip eff.
edtmin_ocn = 0.25,   ! Minimum Precipitation Efficiency
```

```
edtmax_ocn = 0.50,   ! Maximum Precipitation Efficiency
edtmino_ocn = 0.25,  ! Minimum Precipitation Efficiency (o var)
edtmaxo_ocn = 0.50,  ! Maximum Precipitation Efficiency (o var)
edtminx_ocn = 0.25,  ! Minimum Precipitation Efficiency (x var)
edtmaxx_ocn = 0.50,  ! Maximum Precipitation Efficiency (x var)
pbcmax = 150.0,      ! Max depth (mb) of stable layer b/twn LCL & LFC
mincld = 150.0,      ! Min cloud depth (mb).
htmin = -250.0,      ! Min convective heating
htmax = 500.0,       ! Max convective heating
skbmax = 0.4,        ! Max cloud base height in sigma
dtauc = 30.0,        ! Fritsch & Chappell (1980) ABE Removal Timescale (min)
/

&emanparam
minsig = 0.95,   ! Lowest sigma level from which convection can originate
elcrit = 0.0011, ! Autoconversion threshold water content (g/g)
tlcrit = -55.0,  ! Below tlcrit auto-conversion threshold is zero
entp = 1.5,      ! Coefficient of mixing in the entrainment formulation
sigd = 0.05,     ! Fractional area covered by unsaturated dndraft
sigs = 0.12,     ! Fraction of precipitation falling outside of cloud
omtrain = 50.0,  ! Fall speed of rain (Pa/s)
omtsnow = 5.5,   ! Fall speed of snow (Pa/s)
coeffr = 1.0,    ! Coefficient governing the rate of rain evaporation
coeffs = 0.8,    ! Coefficient governing the rate of snow evaporation
cu = 0.7,        ! Coefficient governing convective momentum transport
betae = 10.0,    ! Controls downdraft velocity scale
dtmax = 0.9,     ! Max negative parcel temperature perturbation below LFC
alphae = 0.2,    ! Controls the approach rate to quasi-equilibrium
damp = 0.1,      ! Controls the approach rate to quasi-equilibrium
/
```

Things you need to know here:

1. In case of the mixed schemes 98, 99, both the stanzas are read in. Note in this case for Grell scheme only the relevant (Ocean or Land) control values are used.

2. Minimum and maximum values of the fraction of reevaporated water in the downdraft for the Grell scheme is essentially a measure of the precipitation efficiency: increasing their value generally decrease convective precipitation.

3. Again, read carefully the Reference Manual before attempting any tuning, and report in any work modification of this parameters.

6.1.16 chemparam stanza

This stanza controls the optical active aerosols scheme in the RegCM model. [2]

```
&chemparam
idirect      =    1, ! enable or not aerosol feedbacks on radiation and
                     ! dynamics (aerosol direct and semi direct effcts):
                     !  0 = no coupling. Aerosol are only transported and
                     !      don't interact with radiation scheme.
                     !  1 = no coupling to dynamic and thermodynamic. However
                     !      the clear sky surface and top of atmosphere
                     !      aerosol radiative forcings are diagnosed.
                     !  2 = allows aerosol feedbacks on radiative,
                     !      thermodynamic and dynamic fields.
```

[2] In the future model version a more complete chemical scheme will be introduced

```
ichremlsc    =    1, ! 1 = allows tracer removal (wet deposition) by large
                     !     scale cloud
ichremcvc    =    1, ! 1 = allows tracer removal by convective clouds
ichdrdepo    =    1, ! 1 = enable tracer surface dry deposition. For dust,
                     !     it is calculated by a size settling and dry
                     !     deposition scheme. For other aerosol,a dry
                     !     deposition velocity is simply prescribed further.
                     !     Next release will include an improved aerosol dry
                     !     deposition scheme for non dust aerosols.
ichcumtra    =    1, ! 1 = enable tracer convective transport and mixing.
inpchtrname  =  'DUST','DUST,'BC_HB','BC_HL',
                     ! Tracer identifier. The number of input should be equal
                     ! to ntr you have the choice between:
                     ! DUST  = Dust particle from soil
                     ! BC_HB = Hydrophobic Black carbon aerosol
                     ! BC_HL = Hydrophilic or aged black carbon
                     ! OC_HB = Hydrophobic organic carbon aerosl
                     ! OC_HL = Hydrophilic or aged organic carbon
                     ! SO2   = sulfur dioxide
                     ! SO4   = sulfate aerosol
inpchtrsol   =  0.1, 0.1, 0.05, 0.8,
                     ! Tracer solubility (fraction). The number of input
                     ! should be equal to ntr. Will determine if tracer are
                     ! efficiently removed by wet deposition or not
inpchtrdpv   =  0.,0.,0.00025,0.00025,  0.,0.,0.00025,0.00250,
                     ! Dry deposition velocity (in m/s) over land (first ntr)
                     ! and ocean (second ntr values), a total of ntr*2 values.
                     ! Should be consistent with tracer identifier.
                     ! for DUST type this value is not effectively considered
                     ! since a dry deposition scheme is explicitely included
                     ! in RegCM.
inpdustbsiz  =   0.1,  1., 1., 2.5,
                     ! Lower Size limit (first nbin) and Upper Size limit
                     ! (second nbin values) of diameter bin classes for dust
                     ! (in micrometer). Should never exceed nbin * 2 values.
                     ! So in this example there are two bins of
                     !   *  0.1 - 1.0 micrometer
                     !   *  1.0 - 2.5 micrometer
/
```

Things you need to know here:

1. Always doublecheck consistency in dimensions specified in aerosolparam at 6.1.3 and the number of elements in input arrays here.

2. This stanza is not considered if `ichem` in physicsparam at 6.1.13 is not set to 1.

3. Dust optical properties have been calculated for 4 defaults size bins in RegCM. If you want to modify the bin size for dust / climate feedback interactions consider extending this by yourself. Current bins are $0.01 - 1.00$, $1.00 - 2.50$, $2.50 - 5.00$ and $5.00 - 20.0$ micron diameter.

6.2 The BAND and the CLM options

We will now discuss from the user point of view how to use the two model setups which need to be activated at configure stage.

6.2.1 BAND option

The BAND option if activated allows the user to run a simulation over a tropical band symmetric around the equator. The executable of the model is different in the case of the band, and is named `regcmMPI_band` or `regcmSerial_band`. Note that due to the computational need of the `BAND` model, it is strongly suggested to run it on parallel machines. [3]

Enable

At configure stage (see 3.2.2), input 1 instead of the default 0.

```
****  Do you want to run the tropical band version, BAND - 1 or not - 0?
      [0]
      BAND = 1
```

This will enable a preprocessing flag, and build a different model executable. Note that no modifications are needed for any other part of the model.

Prepare and run

In the case of `BAND` run, the `geoparam` stanza described above in 6.1.2 is mostly ignored, as the projection is set to Normal Mercator, the center of the projection is set to `clat = 0.0`, `clon = 180.0`, and the grid point resolution is calculated as:

$$\frac{2 * \pi * 6370.0}{jx} \tag{6.4}$$

The only parameter you need to set for a `BAND` run is the `i_band` value: set it to 1.

No special modification in model run is required, all steps are equal as in chapter 5. Just substitute the executable name:

```
$> mpirun -np 2 ./Bin/regcmMPI_band band.in
```

Some notes:

1. The model using the `BAND` option is heavy, as the number of points is usually huge to obtain a good horizontal resolution. Check any memory limit is disabled on your platform before attempting a run with the `BAND` option active.
2. The model with the `BAND` option scales very well on a cluster with a large number of processors.

6.2.2 CLM option

The `CLM` option if activated allows the user to run a simulation using the CLM surface model instead of the default BATS1e model. We will not here go in deep in the difference between the two models, read the Reference Manual for this. The executable of the model is different in the case of the `CLM`, and is named `regcmMPI_clm`. Note that in the CLM case only the MPI enabled compilation is supported (no serial), and no subgridding is possible (`nsg` is always 1).

[3] The serial option will not be supported in future releases

Enable

At configure stage (see 3.2.2), input 1 instead of the default 0.

```
****  Do you want to enable CLM  - 1 or not - 0?
      [0]
      CLM = 1
```

This will enable a preprocessing flag, and build a different model executable. Note that no modifications are needed for any other part of the model, but this triggers the building of another pre-processing program, `clm2rcm`.

Prepare and run

The `CLM` configuration requires a separate stanza in the namelist input file.

```
&clmparam
dirclm = 'input/', ! CLM path to Input data produced by clm2rcm. If
                   ! relative, It should be how to reach the Input dir
                   ! from the Run dir.
clmfrq =  12.,     ! Frequency for CLM own output write
imask  =   1,      ! For CLM, Type of land surface parameterization
                   !   1 => using DOMAIN.INFO for landmask (same as BATS)
                   !   2 => using mksrf_navyoro file landfraction for
                   !        landmask and perform a weighted average over
                   !        ocean/land gridcells; for example:
                   ! tgb = tgb_ocean*(1-landfraction)+tgb_land*landfraction
/
```

Things you need to know here:

1. The `inpter` path defined in `terrainparam` stanza described in 6.1.4 is used also by the `clm2rcm` program. See at 4.4 how to obtain needed datasets.
2. The file `pft-physiology.c070207` should be manually copied in the `dirclm` directory before running the model.
3. The clmfrq is relative to the output produced by the `CLM` model itself, and does not control the RegCM model output. To know the `CLM` output file content, refer to CLM 3.5 documentation.
4. The `imask = 2` option cannot be used with the `icup` cumulus convection schemes $2, 98, 99$, which rely on the BATS1e landmask.

In the case of `CLM` run, the user needs to run, after the `terrain` program, the `clm2rcm` program, and copy the `pft-physiology.c070207` in the input directory:

```
$> cd $REGCM_RUN
$> ./Bin/terrain regcm.in
$> ./Bin/clm2rcm regcm.in
$> cp $REGCM_GLOBEDAT/CLM/pft-physiology.c070207 input/
```

The `clm2rcm` program interpolates global land characteristics datasets to the RegCM projected grid. The content of the `pft-physiology.c070207` file are described in the `pft-physiology.c070207.readme` file. All the other pre-processing steps are just equal to the one detailed in chapter 5. To run the `CLM` option in the RegCM model, just substitute the executable name:

```
$> mpirun -np 2 ./Bin/regcmMPI_clm regcm.in
```

Note that the `CLM` land model is much heavier than the BATS1e model, and computing time increases.

6.3 Sensitivity experiments hint

Although the LBC forcing does provide a constraint for the model, as any RCM, RegCM4 is characterized by a certain level of internal variability due to its nonliner processes (e.g. convection).

For example, if small perturbations are introduced in the initial or lateral boundary conditions, the model will generally produce different patterns of, e.g. precipitation, that appear as (sometimes seemingly organized) noise when compared to the control simulation.

This noise depends on domain size and climatic regimes, for example it is especially pronounced in warm climate regimes (e.g. tropics or during the summer season) and large doamins.

When doing for example sensitivity experiments to model modifications, e.g. to land use change, this internal variability noise can be misinterpreted as a model response to the factor modified.

Users of RegCM4 should be aware of this when they do sensitivity experiments. The best way to filter out this noise is to perform ensembles of simulations and lok at the ensemble averages to extract the real model response from the noise.

Chapter 7

Postprocessing tools

The new netCDF output format allows users to use a number of general purpose tools to postprocess model output files. We will in this section do a quick review of some of the Open Source and Free Software ones.

7.1 Command line tools

Three major set of tools may help you do even complex calculation just from command line prompt.

7.1.1 netCDF library tools

The netCDF library itself offers three basic tools to play with netCDF archived data.

- `ncdump` program, generates a text representation of a specified netCDF file on standard output. The text representation is in a form called CDL (network Common Data form Language) that can be viewed, edited, or serve as input to `ncgen`, thus `ncdump` and `ncgen` can be used as inverses to transform data representation between binary and text representations. `ncdump` may also be used as a simple browser for netCDF datasets, to display the dimension names and lengths; variable names, types, and shapes; attribute names and values; and optionally, the values of data for all variables or selected variables in a netCDF dataset. Sample usage patterns:

  1. Look at the structure of the data in the netCDF dataset:

     ```
     ncdump -c test_001_SRF.1990060100.nc
     ```

  2. Produce a fully-annotated (one data value per line) listing of the data for the variables time and t2m, using FORTRAN conventions for indices, and show the floating-point data with only four significant digits of precision and the time values with ISO format:

     ```
     ncdump -v time,t2m -p 4 -t -f \
              fortran test_001_SRF.1990060100.nc
     ```

- `ncgen` program, the reverse of the `ncdump` program: generates a netCDF file or a C or FORTRAN program that creates a netCDF dataset from a CDL input. Sample usage patterns:

  1. From a CDL file, generate a binary netCDF file:

     ```
     ncgen -o test_001_SRF.1990060100_modif.nc \
              test_001_SRF.1990060100.cdl
     ```

  2. From a CDL file, generate a Fortran 77 program to write the netCDF file:

     ```
     ncgen -f test_001_SRF.1990060100.cdl > prog.f
     ```

- `nccopy` utility copies an input netCDF file to an output netCDF file, in any of the four format variants, if possible, and in function of the selected output format add compression filter and/or data chunking. Sample usage patterns:

  1. Convert a netCDF dataset to a netCDF 4 classic model compressed data file using shuffling to enhance compression level:

     ```
     nccopy -k 4 -d 9 -s test_001_SRF.1990060100.nc \
              test_001_SRF.1990060100_compressed.nc
     ```

You can also find, in the `Tools/Programs/RegCM_read` directory under `$REGCM_ROOT` a sample program to read an output file using the netCDF library you can modify to fit your needs. Another sample program to read SAV output file format can be found under the directory `SAV_read` in the `Tools/Programs` directory.

7.1.2 NetCDF operators NCO

This set of tools can be considered a swiss army knife to manage netCDF datasets. There are multiple operators, and Each operator takes netCDF files as input, then operates (e.g., derives new data, averages, hyperslabs, manipulates metadata) and produces a netCDF output file. The single-command style of NCO allows users to manipulate and analyze files interactively, or with simple scripts that avoid some overhead of higher level programming environments. The major tools are:

- `ncap2` netCDF Arithmetic Processor
- `ncatted` netCDF Attribute Editor
- `ncbo` netCDF Binary Operator
- `ncea` netCDF Ensemble Averager
- `ncecat` netCDF Ensemble Concatenator
- `ncflint` netCDF File Interpolator
- `ncks` netCDF Kitchen Sink
- `ncpdq` netCDF Permute Dimensions Quickly, Pack Data Quietly

- `ncra` netCDF Record Averager
- `ncrcat` netCDF Record Concatenator
- `ncrename` netCDF Renamer
- `ncwa` netCDF Weighted Averager

A comprehensive user guide can be found at:

`http://nco.sourceforge.net/nco.html`

Sample usage patterns:

1. Get value of t2m variable at a particular point for all timesteps with a prescribed format one per line on stdout:

```
ncks -C -H -s "%6.2f\n" -v t2m -d iy,16 -d jx,16 \
         test_001_SRF.1990060100.nc
```

2. Extract one timestep of t2m from a file and save into a new netCDF file:

```
ncks -c -v t2m -d time,6 test_001_SRF.1990060100.nc \
       test_001_SRF.1990060212.nc
```

3. Cat together a year worth of output data for the single t2m variable into a single file:

```
ncrcat -c -v t2m test_001_SRF.1990??0100.nc \
         test_001_T2M.1990.nc
```

4. Get the DJF mean value of the tempertaure from a multiyear run:

```
ncra -c -v t2m test_001_SRF.????120100.nc \
               test_001_SRF.????010100.nc \
               test_001_SRF.????020100.nc \
               test_001_DJF_T2M.nc
```

We strongly encourage you to read the on-line user guide of the NCO tools. You will for sure get a boost on your data manipulation and analysis skills.

7.1.3 Climate data Operators CDO

The monolithic `cdo` program from the Max Planck Institut f´ur Meteorologie implements a really comprehensive collection of command line Operators to manipulate and analyse Climate and NWP model Data either in netCDF or GRIB format. There are more than 400 operators available, covering the following topics:

- File information and file operations
- Selection and Comparision
- Modification of meta data
- Arithmetic operations

- Statistical analysis
- Regression and Interpolation
- Vector and spectral Transformations
- Formatted I/O
- Climate indices

We wont make here a comprehensive analysis of this tool, but you can find some ideas in the `PostProc` directory on `$REGCM_ROOT` reading the two sample `average` and `regrid` scripts, which use a combination of `NCO` programs and `cdo` operators to reach goal. A very simple usage pattern for example to obtain a monthly mean is:

```
cdo monmean test_001_T2M.1990.nc
```

7.2 GrADS program

This tool is the one mostly used at ICTP to analyze and plot model output results. It can be used either as an interactive tool either as a batch data analysis tool. We have already written in chapter 5 about the helper program `GrADSNcPlot` which can be used to interactively plot model output results. We will here detail why an helper program is needed and how it does work. For information regarding the `grads` program itself, a comprehensive guide may be found at:

```
http://www.iges.org/grads/gadoc/users.html
```

7.2.1 GrADS limits

The `grads` program is powerful, yet has limits:

1. Only the equirectangular projection or Plate Carrée is supported. Some other projections can be used through a pdef entry in the `CTL` file using the internal direct preprojection engines, but not all RegCM supported projections are supported using direct engine.
2. NetCDF format allows multidimensional variables, while `grads` supports just four dimensional (time,level,latitude,longitude) variables.

Luckily, these limits can be exceeded, carefully telling `grads` the RegCM data structure using the `CTL` file and one ancillary `proj` file:

1. The `grads` program allows usage of the `pdef BILIN` option in the `CTL` file, which allows the user to specify a supplementary file name. In this file are stored three lat-lon floating-point grids which have for each point on the equirectangular grid the indexes i,j on the projected grid, as well as wind rotation values.
2. The `grads` program allows identifying four dimentional slices of a multidimensional variable as new variables, providing them a unique name. This is how we are able to see in `grads` chemical output variables.

While the `GrADSNcPlot` program allows interactive plotting and after quitting the `grads` program removes the `CTL` file and the `proj` file, the `GrADSNcPrepare` program only creates this two files, allowing share of the `proj` file between multiple `CTL` files sharing the same RegCM domain (i.e. it creates just only once the `proj` file). To use the `grads` program, you need to have both this ancillary files together with the data netCDF file.

A collection of sample `grads` scripts commonly used at ICTP to plot simulation results can be found in the `Tools/Scripts/GrADS` directory under `$REGCM_ROOT`.

7.3 CISL's NCL : NCAR Command Language

This awesome tool from NCAR is an interpreted language designed for scientific data analysis and visualization. Noah Diffenbaugh and Mark Snyder have created a website dedicated to visualizing RegCM3 output using the NCAR Command Language (NCL). These scripts where built using RegCM3 model output converted to netCDF using an external converter. They have been adapted to serve as very basic example scripts to process a native RegCM 4.1 output data file or do some data analysis using the `NCL` language and are available in the `Tools/Scripts/NCL/examples` directory. Travis O'Brien from the User Community also contributed sample scripts, which may be found under the `Tools/Scripts/NCL` directory.

7.4 R Statistical Computing Language

The `R` statistical computing language is able with an add on package to load into interal data structure a meteorological field read from a netCDF RegCM output. A sample script to load and plot the 2m Temperature at a selected timestep can be used as a reference to develop a real powerful statistical analysis of model results: it is under `Tools/Scripts/R`.

7.5 Non free tools

Note that the netCDF format, using plugins or native capabilities, allows clean access to model output from a number of non free tools like Matlab^{TM}or IDL^{TM}.

For a more complete list of tools, you are invited to scroll down the very long list of tools at:

http://www.unidata.ucar.edu/software/netcdf/software.html

Chapter 8

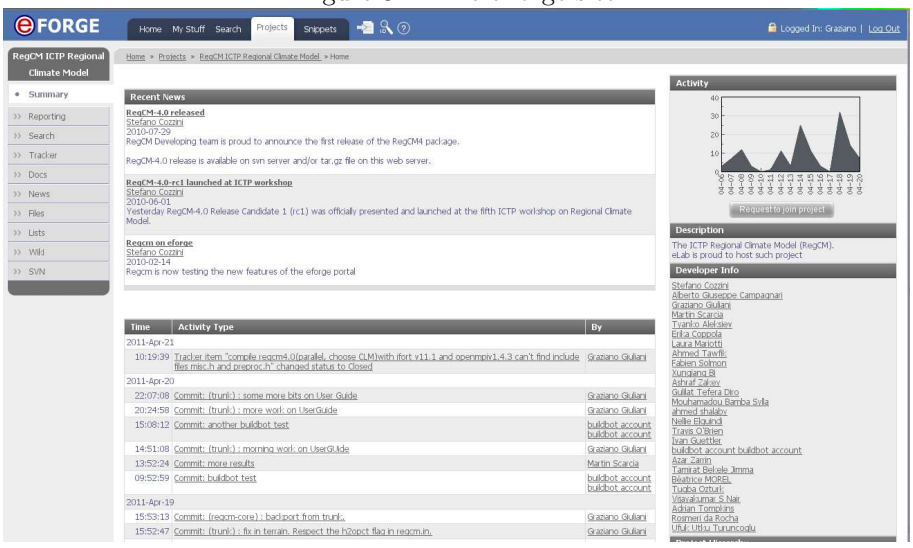
Getting help and reporting bugs

8.1 The eforge site

A new welcoming home for the RegCM Community has been built with the help of Italian National Research Council CNR Democritos Group on the e-science Lab E-Forge web site:

https://eforge.escience-lab.org/gf/project/regcm

Figure 8.1: The e-forge site



On this site you have access with a simple registration to a friendly bug tracking system under the *tracker* link, allowing the users to post problems and bugs they discover.

It allows posting also of files to give you the opportunity to provide as much information as possible about the environment the model is running at your institution, helping us better understand and solve efficiently your problems.

Help us grow the model to fit your requirements, giving the broader user community the benefit of a valuable tool to do better research.

A few general rules, adapted from :

`http://www.chiark.greenend.org.uk/~sgtatham/bugs.html`

- The first aim of a bug report is to let us see the failure with our own eyes. So give us detailed instructions so that we can make it fail for ourself.
- In case the first aim doesn't succeed, and the programmer can't see it failing themselves, the second aim of a bug report is to describe what went wrong. Describe everything in detail. State what you saw, and also state what you expected to see. Write down the error messages, especially if they have numbers in.
- By all means try to diagnose the fault yourself if you think you can, but if you do, you should still report the symptoms as well.
- Be ready to provide extra information if the programmer needs it. If they didn't need it, they wouldn't be asking for it. They aren't being deliberately awkward. Have version numbers at your fingertips, because they will probably be needed.
- Write clearly. Say what you mean, and make sure it can't be misinterpreted.
- Above all, be precise.

Specific Rules for RegCM package:

1. Use the tracker system and give us details about which part of the code failed (terrain/icbc/regmc/postproc etc)
2. Give us information about compiler you are using
3. Give us information about the version you are using
4. Provide us with your input file so we can test/repeat your problem
5. Be sure to have read the documentation before posting/asking for help
6. Be patient: not always there is somebody ready to answer your question..

Chapter 9

Appendices

We will review here a sample installation session of software needed to install the RegCM model.

The starting point is here a Linux system on a multicore processor box, and the final goal is to have an optimized system to run the model. I will use `bash` as my shell and assume that GNU development tools like `make`, `sed`, `awk` are installed as part of the default Operating System environment as is the case in most Linux distro. I will require also for commodity a command line web downloader such as `curl` installed on the system, along its development libraries to be used to enable OpenDAP remote data access protocol capabilities of netCDF library. Standard file management tools such as `tar` and `gzip` are also required. The symbol `$>` will stand for a shell prompt. I will assume that the process is performed as a normal system user, which will own all the toolchain. I will be now just the `regcm` user.

9.1 Identify Processor

First step is to identify the processor to know its capabilities:

```
$> cat /proc/cpuinfo
```

This command will ask to the operating system to print processor informations. A sample answer on my laptop is:

```
processor       : 0
vendor_id       : GenuineIntel
cpu family      : 6
model           : 30
model name      : Intel(R) Core(TM) i7 CPU       Q 740  @ 1.73GHz
stepping        : 5
cpu MHz         : 933.000
cache size      : 6144 KB
physical id     : 0
siblings        : 8
core id         : 0
cpu cores       : 4
```

```
apicid          : 0
initial apicid  : 0
fpu             : yes
fpu_exception   : yes
cpuid level     : 11
wp              : yes
flags           : fpu vme de pse tsc msr pae mce cx8 apic sep mtrr pge
mca cmov pat pse36 clflush dts acpi mmx fxsr sse sse2 ss ht tm pbe syscall
nx rdtscp lm constant_tsc arch_perfmon pebs bts rep_good nopl xtopology
nonstop_tsc aperfmperf pni dtes64 monitor ds_cpl vmx smx est tm2 ssse3
cx16 xtpr pdcm sse4_1 sse4_2 popcnt lahf_lm ida dts tpr_shadow vnmi
flexpriority ept vpid
bogomips        : 3467.81
clflush size    : 64
cache_alignment : 64
address sizes   : 36 bits physical, 48 bits virtual
power management:
```

repeated eight time with Processor Ids from 0 to 7: I have a Quad Core Intel with Hyperthreading on (this multiply by 2 the reported processor list). The processor reports here also to support Intel Streaming SIMD Extensions V4.2, which can be later used to speed up code execution vectorizing floating point operation on any single CPU core.

9.2 Chose compiler

Depending on the processor, we can chose which compiler to use. On a Linux box, we have multiple choices:

- GNU Gfortran
- G95
- Intel ifort compiler
- Portland pgf90 compiler
- Absoft ProFortran
- NAG Fortran Compiler

and for sure other which I may not be aware of. All of these compilers have pros and cons, so I am just for now selecting one in the pool only to continue the exposition. I am not selecting the trivial solution of Gfortran as most Linux distributions have it already packaged, and all the other required software as well (most complete distribution I am aware of for this is Fedora: all needed software is packaged and it is a matter of yum install).

So let us assume I have licensed the Intel Composer XE Professional Suite 12.0.2 on my laptop. My system administrator installed it on the default location under `/opt/intel`, and I have my shell environment update loading vendor provided script:

```
$> source /opt/intel/bin/compilervars.sh intel64
```

With some modification (the path, the script, the arguments to the script), same step is to be performed for all non-GNU compilers in the above list, and is documented in the installation manual of the compiler itself.

In case of Intel, to check the correct behaviour of the compiler, try to type the following command:

```
$> ifort --version
ifort (IFORT) 12.0.2 20110112
Copyright (C) 1985-2011 Intel Corporation.  All rights reserved.
```

I am skipping here any problem that may arise from license installation for any of the compilers, so I am assuming that if the compiler is callable, it works. As this step is usually performed by a system administrator on the machine, I am assuming a skilled professional will take care of that.

9.3 Environment setup

To efficiently use the compilers, I will setup now some environment variables. On my system (see the above processor informations) I will use:

```
$> # Where all the software will be installed ?
$> # I am chosing here a place under my home directory.
$> export INTELROOT=/home/regcm/intelsoft
$> export INTELSRC=/home/regcm/intelsoft/src
$> mkdir -p $INTELROOT/{bin,include,lib,share/man,src}
$> # the C compiler. I am assuming here to have the whole Intel
$> # Composer XE suite, so I will use the intel C compiler.
$> export CC=icc
$> # the C++ compiler, the intel one.
$> export CXX=icpc
$> # the Fortran 9X compiler.
$> export FC=ifort
$> # the Foirtran 77 compiler. For intel, is just the fortran one.
$> export F77=ifort
$> # C Compiler flags
$> export CFLAGS="-O3 -xHost -axSSE4.2 -fPIC"
$> # F9X Compiler flags
$> export FCFLAGS="-O3 -xHost -axSSE4.2 -fPIC"
$> # F77 Compiler flags
$> export FFLAGS="-O3 -xHost -axSSE4.2 -fPIC"
$> # CXX Compiler flags
$> export CXXFLAGS="-O3 -xHost -axSSE4.2 -fPIC"
$> # Linker flags
$> export LDFLAGS="-Wl,-rpath=$INTELROOT/lib \
> -Wl,-rpath=/opt/intel/lib/intel64 -i-dynamic"
$> # Preset PATH to use the installed software during build
$> export PATH=$INTELROOT/bin:$PATH
```

```
$> export LD_LIBRARY_PATH=$LD_LIBRARY_PATH:$INTELROOT/lib
$> export MANPATH=$INTELROOT/share/man:$MANPATH
```

This step will allow me not to specify those variables at every following step. Depending on the above compiler selected, those flags my differ for you, but the concept is that I am selecting a performance target build for the machine I am on. I am now ready to compile software.

9.4 Compression Library Installation

To have a complete optimized stack, I will compile also here the compression libraries optionally needed by HDF5 library.

I will need the sources:

```
$> cd $INTELSRC
$> curl -o zlib-1.2.5.tar.gz http://zlib.net/zlib-1.2.5.tar.gz
$> curl -o szip-2.1.tar.gz \
> http://www.hdfgroup.org/ftp/lib-external/szip/2.1/src/szip-2.1.tar.gz
```

Please note that the szip library license allows to use that software only together with HDF5 library.

Then I must decompress the sources:

```
$> cd $INTELSRC
$> tar -zxvf zlib-1.2.5.tar.gz
$> tar -zxvf szip-2.1.tar.gz
```

Let us start with zlib:

```
$> cd $INTELSRC
$> cd zlib-1.2.5
$> LDSHARED="icc -shared -Wl,-soname \
> -Wl,libz.so.1,--version-script,zlib.map" \
> ./configure --prefix=$INTELROOT
$> make
$> make check
$> make install
```

This will install the zlib software under `/home/regcm/intelsoft`. Let us continue with szip.

```
$> cd $INTELSRC
$> cd szip-2.1
$> ./configure --prefix=$INTELROOT
$> make
$> make check
$> make install
```

This will install the szip software under `/home/regcm/intelsoft`.

9.5 HDF5 Library installation

Download the source pack:

```
$> cd $INTELSRC
$> curl -o hdf5-1.8.6.tar.bz2 \
> ftp://ftp.hdfgroup.org/HDF5/current/src/hdf5-1.8.6.tar.bz2
```

Install it:

```
$> cd $INTELSRC
$> cd hdf5-1.8.6
$> ./configure --prefix=$INTELROOT --enable-hl --enable-linux-lfs \
> --enable-production --with-pic --docdir=$INTELROOT/share/doc/hdf5/ \
> --with-szlib=$INTELROOT --with-zlib=$INTELROOT
$> make
$> make check
$> make install
```

9.6 netCDF Library installation

Download the source pack:

```
$> cd $INTELSRC
$> curl -o netcdf-4.1.2.tar.gz \
> http://www.unidata.ucar.edu/downloads/netcdf/ftp/netcdf-4.1.2.tar.gz
```

Install it:

```
$> cd $INTELSRC
$> cd netcdf-4.1.2
$> ./configure --prefix=$INTELROOT --enable-shared --enable-netcdf-4 \
> --with-udunits --with-libcf --enable-dap-netcdf --enable-cxx-4 \
> --with-hdf5=$INTELROOT --with-zlib=$INTELROOT --with-szlib=$INTELROOT
$> make
$> make check
$> make install
```

9.7 OpenMPI library installation

This optional step will install OpenMPI message passing library to enable parallel run of the RegCM model using all cores of my processor. Download the souece pack:

```
$> cd $INTELSRC
$> curl -o openmpi-1.4.3.tar.bz2 \
> http://www.open-mpi.org/software/ompi/v1.5/downloads/openmpi-1.4.3.tar.bz2
```

Install it:

```
$> cd $INTELSRC
$> cd openmpi-1.4.3
$> ./configure --prefix=$INTELROOT --sysconfdir=$INTELROOT/etc/openmpi \
> --mandir=$INTELROOT/share/man --libdir=$INTELROOT/lib --enable-mpi-f90
$> make
$> make check
$> make install
```

9.8 Final step

To enable all the ready installed software to be used by the regcm user whenever it logs in the box, edit the `.bashrc` file in the home directory and add the following lines:

```
export PATH=$INTELROOT/bin:$PATH
export LD_LIBRARY_PATH=$LD_LIBRARY_PATH:$INTELROOT/lib
export MANPATH=$INTELROOT/share/man:$MANPATH
```

Bibliography

Giorgi, F., Regcm version 4.1 reference manual, *Tech. rep.*, ICTP Trieste, 2011.

R. Courant, K. F., and H. Lewy, über die partiellen differenzengleichungen der mathematischen physik, *Mathematische Annalen*, *100*(1), 3274, 1928.

Rew, R. K., and G. P. Davis, Netcdf: An interface for scientific data access, *IEEE Computer Graphics and Applications*, *10*(4), 76–82, 1990.

GNU Free Documentation License
Version 1.3, 3 November 2008

0. PREAMBLE

The purpose of this License is to make a manual, textbook, or other functional and useful document "free" in the sense of freedom: to assure everyone the effective freedom to copy and redistribute it, with or without modifying it, either commercially or noncommercially. Secondarily, this License preserves for the author and publisher a way to get credit for their work, while not being considered responsible for modifications made by others.

This License is a kind of "copyleft", which means that derivative works of the document must themselves be free in the same sense. It complements the GNU General Public License, which is a copyleft license designed for free software.

We have designed this License in order to use it for manuals for free software, because free software needs free documentation: a free program should come with manuals providing the same freedoms that the software does. But this License is not limited to software manuals; it can be used for any textual work, regardless of subject matter or whether it is published as a printed book. We recommend this License principally for works whose purpose is instruction or reference.

1. APPLICABILITY AND DEFINITIONS

This License applies to any manual or other work, in any medium, that contains a notice placed by the copyright holder saying it can be distributed under the terms of this License. Such a notice grants a world-wide, royalty-free license, unlimited in duration, to use that work under the conditions stated herein. The "Document", below, refers to any such manual or work. Any member of the public is a licensee, and is addressed as "you". You accept the license if you copy, modify or distribute the work in a way requiring permission under copyright law.

A "Modified Version" of the Document means any work containing the Document or a portion of it, either copied verbatim, or with modifications and/or translated into another language.

A "Secondary Section" is a named appendix or a front-matter section of the Document that deals exclusively with the relationship of the publishers or authors of the Document to the Document’s overall subject (or to related matters) and contains nothing that could fall directly within that overall subject. (Thus, if the Document is in part a textbook of mathematics, a Secondary Section may not explain any mathematics.) The relationship could be a matter of historical connection with the subject or with related matters, or of legal, commercial, philosophical, ethical or political position regarding them.

The "Invariant Sections" are certain Secondary Sections whose titles

are designated, as being those of Invariant Sections, in the notice that says that the Document is released under this License. If a section does not fit the above definition of Secondary then it is not allowed to be designated as Invariant. The Document may contain zero Invariant Sections. If the Document does not identify any Invariant Sections then there are none.

The "Cover Texts" are certain short passages of text that are listed, as Front-Cover Texts or Back-Cover Texts, in the notice that says that the Document is released under this License. A Front-Cover Text may be at most 5 words, and a Back-Cover Text may be at most 25 words.

A "Transparent" copy of the Document means a machine-readable copy, represented in a format whose specification is available to the general public, that is suitable for revising the document straightforwardly with generic text editors or (for images composed of pixels) generic paint programs or (for drawings) some widely available drawing editor, and that is suitable for input to text formatters or for automatic translation to a variety of formats suitable for input to text formatters. A copy made in an otherwise Transparent file format whose markup, or absence of markup, has been arranged to thwart or discourage subsequent modification by readers is not Transparent. An image format is not Transparent if used for any substantial amount of text. A copy that is not "Transparent" is called "Opaque".

Examples of suitable formats for Transparent copies include plain ASCII without markup, Texinfo input format, LaTeX input format, SGML or XML using a publicly available DTD, and standard-conforming simple HTML, PostScript or PDF designed for human modification. Examples of transparent image formats include PNG, XCF and JPG. Opaque formats include proprietary formats that can be read and edited only by proprietary word processors, SGML or XML for which the DTD and/or processing tools are not generally available, and the machine-generated HTML, PostScript or PDF produced by some word processors for output purposes only.

The "Title Page" means, for a printed book, the title page itself, plus such following pages as are needed to hold, legibly, the material this License requires to appear in the title page. For works in formats which do not have any title page as such, "Title Page" means the text near the most prominent appearance of the work’s title, preceding the beginning of the body of the text.

The "publisher" means any person or entity that distributes copies of the Document to the public.

A section "Entitled XYZ" means a named subunit of the Document whose title either is precisely XYZ or contains XYZ in parentheses following text that translates XYZ in another language. (Here XYZ stands for a specific section name mentioned below, such as "Acknowledgements", "Dedications", "Endorsements", or "History".) To "Preserve the Title" of such a section when you modify the Document means that it remains a section "Entitled XYZ" according to this definition.

The Document may include Warranty Disclaimers next to the notice which states that this License applies to the Document. These Warranty Disclaimers are considered to be included by reference in this License, but only as regards disclaiming warranties: any other implication that these Warranty Disclaimers may have is void and has no effect on the meaning of this License.

2. VERBATIM COPYING

You may copy and distribute the Document in any medium, either commercially or noncommercially, provided that this License, the copyright notices, and the license notice saying this License applies to the Document are reproduced in all copies, and that you add no other conditions whatsoever to those of this License. You may not use technical measures to obstruct or control the reading or further copying of the copies you make or distribute. However, you may accept compensation in exchange for copies. If you distribute a large enough number of copies you must also follow the conditions in section 3.

You may also lend copies, under the same conditions stated above, and you may publicly display copies.

3. COPYING IN QUANTITY

If you publish printed copies (or copies in media that commonly have printed covers) of the Document, numbering more than 100, and the Document's license notice requires Cover Texts, you must enclose the copies in covers that carry, clearly and legibly, all these Cover Texts: Front-Cover Texts on the front cover, and Back-Cover Texts on the back cover. Both covers must also clearly and legibly identify you as the publisher of these copies. The front cover must present the full title with all words of the title equally prominent and visible. You may add other material on the covers in addition. Copying with changes limited to the covers, as long as they preserve the title of the Document and satisfy these conditions, can be treated as verbatim copying in other respects.

If the required texts for either cover are too voluminous to fit legibly, you should put the first ones listed (as many as fit reasonably) on the actual cover, and continue the rest onto adjacent pages.

If you publish or distribute Opaque copies of the Document numbering more than 100, you must either include a machine-readable Transparent copy along with each Opaque copy, or state in or with each Opaque copy a computer-network location from which the general network-using public has access to download using public-standard network protocols a complete Transparent copy of the Document, free of added material. If you use the latter option, you must take reasonably prudent steps, when you begin distribution of Opaque copies in quantity, to ensure that this Transparent copy will remain thus accessible at the stated location until at least one year after the last time you distribute an Opaque copy (directly or through your agents or retailers) of that edition to the public.

It is requested, but not required, that you contact the authors of the Document well before redistributing any large number of copies, to give them a chance to provide you with an updated version of the Document.

4. MODIFICATIONS

You may copy and distribute a Modified Version of the Document under the conditions of sections 2 and 3 above, provided that you release the Modified Version under precisely this License, with the Modified Version filling the role of the Document, thus licensing distribution and modification of the Modified Version to whoever possesses a copy of it. In addition, you must do these things in the Modified Version:

A. Use in the Title Page (and on the covers, if any) a title distinct from that of the Document, and from those of previous versions (which should, if there were any, be listed in the History section of the Document). You may use the same title as a previous version if the original publisher of that version gives permission.
B. List on the Title Page, as authors, one or more persons or entities responsible for authorship of the modifications in the Modified Version, together with at least five of the principal authors of the Document (all of its principal authors, if it has fewer than five), unless they release you from this requirement.
C. State on the Title page the name of the publisher of the Modified Version, as the publisher.
D. Preserve all the copyright notices of the Document.
E. Add an appropriate copyright notice for your modifications adjacent to the other copyright notices.
F. Include, immediately after the copyright notices, a license notice giving the public permission to use the Modified Version under the terms of this License, in the form shown in the Addendum below.
G. Preserve in that license notice the full lists of Invariant Sections and required Cover Texts given in the Document's license notice.
H. Include an unaltered copy of this License.
I. Preserve the section Entitled "History", Preserve its Title, and add to it an item stating at least the title, year, new authors, and publisher of the Modified Version as given on the Title Page. If there is no section Entitled "History" in the Document, create one stating the title, year, authors, and publisher of the Document as given on its Title Page, then add an item describing the Modified Version as stated in the previous sentence.
J. Preserve the network location, if any, given in the Document for public access to a Transparent copy of the Document, and likewise the network locations given in the Document for previous versions it was based on. These may be placed in the "History" section. You may omit a network location for a work that was published at least four years before the Document itself, or if the original publisher of the version it refers to gives permission.
K. For any section Entitled "Acknowledgements" or "Dedications", Preserve the Title of the section, and preserve in the section all the substance and tone of each of the contributor acknowledgements and/or dedications given therein.
L. Preserve all the Invariant Sections of the Document, unaltered in their text and in their titles. Section numbers or the equivalent are not considered part of the section titles.
M. Delete any section Entitled "Endorsements". Such a section may not be included in the Modified Version.
N. Do not retitle any existing section to be Entitled "Endorsements" or to conflict in title with any Invariant Section.
O. Preserve any Warranty Disclaimers.

If the Modified Version includes new front-matter sections or appendices that qualify as Secondary Sections and contain no material copied from the Document, you may at your option designate some or all of these sections as invariant. To do this, add their titles to the list of Invariant Sections in the Modified Version's license notice. These titles must be distinct from any other section titles.

You may add a section Entitled "Endorsements", provided it contains nothing but endorsements of your Modified Version by various parties--for example, statements of peer review or that the text has been approved by an organization as the authoritative definition of a standard.

You may add a passage of up to five words as a Front-Cover Text, and a passage of up to 25 words as a Back-Cover Text, to the end of the list of Cover Texts in the Modified Version. Only one passage of Front-Cover Text and one of Back-Cover Text may be added by (or through arrangements made by) any one entity. If the Document already includes a cover text for the same cover, previously added by you or by arrangement made by the same entity you are acting on behalf of, you may not add another; but you may replace the old one, on explicit permission from the previous publisher that added the old one.

The author(s) and publisher(s) of the Document do not by this License give permission to use their names for publicity for or to assert or imply endorsement of any Modified Version.

5. COMBINING DOCUMENTS

You may combine the Document with other documents released under this License, under the terms defined in section 4 above for modified versions, provided that you include in the combination all of the Invariant Sections of all of the original documents, unmodified, and list them all as Invariant Sections of your combined work in its license notice, and that you preserve all their Warranty Disclaimers.

The combined work need only contain one copy of this License, and multiple identical Invariant Sections may be replaced with a single copy. If there are multiple Invariant Sections with the same name but different contents, make the title of each such section unique by adding at the end of it, in parentheses, the name of the original author or publisher of that section if known, or else a unique number. Make the same adjustment to the section titles in the list of Invariant Sections in the license notice of the combined work.

In the combination, you must combine any sections Entitled "History" in the various original documents, forming one section Entitled "History"; likewise combine any sections Entitled "Acknowledgements", and any sections Entitled "Dedications". You must delete all sections Entitled "Endorsements".

6. COLLECTIONS OF DOCUMENTS

You may make a collection consisting of the Document and other documents released under this License, and replace the individual copies of this License in the various documents with a single copy that is included in the collection, provided that you follow the rules of this License for verbatim copying of each of the documents in all other respects.

You may extract a single document from such a collection, and distribute it individually under this License, provided you insert a copy of this License into the extracted document, and follow this License in all other respects regarding verbatim copying of that document.

7. AGGREGATION WITH INDEPENDENT WORKS

A compilation of the Document or its derivatives with other separate and independent documents or works, in or on a volume of a storage or distribution medium, is called an "aggregate" if the copyright resulting from the compilation is not used to limit the legal rights

of the compilation's users beyond what the individual works permit. When the Document is included in an aggregate, this License does not apply to the other works in the aggregate which are not themselves derivative works of the Document.

If the Cover Text requirement of section 3 is applicable to these copies of the Document, then if the Document is less than one half of the entire aggregate, the Document's Cover Texts may be placed on covers that bracket the Document within the aggregate, or the electronic equivalent of covers if the Document is in electronic form. Otherwise they must appear on printed covers that bracket the whole aggregate.

8. TRANSLATION

Translation is considered a kind of modification, so you may distribute translations of the Document under the terms of section 4. Replacing Invariant Sections with translations requires special permission from their copyright holders, but you may include translations of some or all Invariant Sections in addition to the original versions of these Invariant Sections. You may include a translation of this License, and all the license notices in the Document, and any Warranty Disclaimers, provided that you also include the original English version of this License and the original versions of those notices and disclaimers. In case of a disagreement between the translation and the original version of this License or a notice or disclaimer, the original version will prevail.

If a section in the Document is Entitled "Acknowledgements", "Dedications", or "History", the requirement (section 4) to Preserve its Title (section 1) will typically require changing the actual title.

9. TERMINATION

You may not copy, modify, sublicense, or distribute the Document except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense, or distribute it is void, and will automatically terminate your rights under this License.

However, if you cease all violation of this License, then your license from a particular copyright holder is reinstated (a) provisionally, unless and until the copyright holder explicitly and finally terminates your license, and (b) permanently, if the copyright holder fails to notify you of the violation by some reasonable means prior to 60 days after the cessation.

Moreover, your license from a particular copyright holder is reinstated permanently if the copyright holder notifies you of the violation by some reasonable means, this is the first time you have received notice of violation of this License (for any work) from that copyright holder, and you cure the violation prior to 30 days after your receipt of the notice.

Termination of your rights under this section does not terminate the licenses of parties who have received copies or rights from you under this License. If your rights have been terminated and not permanently reinstated, receipt of a copy of some or all of the same material does not give you any rights to use it.

10. FUTURE REVISIONS OF THIS LICENSE

The Free Software Foundation may publish new, revised versions of the GNU Free Documentation License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns. See http://www.gnu.org/copyleft/.

Each version of the License is given a distinguishing version number. If the Document specifies that a particular numbered version of this License "or any later version" applies to it, you have the option of following the terms and conditions either of that specified version or of any later version that has been published (not as a draft) by the Free Software Foundation. If the Document does not specify a version number of this License, you may choose any version ever published (not as a draft) by the Free Software Foundation. If the Document specifies that a proxy can decide which future versions of this License can be used, that proxy's public statement of acceptance of a version permanently authorizes you to choose that version for the Document.

11. RELICENSING

"Massive Multiauthor Collaboration Site" (or "MMC Site") means any World Wide Web server that publishes copyrightable works and also provides prominent facilities for anybody to edit those works. A public wiki that anybody can edit is an example of such a server. A "Massive Multiauthor Collaboration" (or "MMC") contained in the site means any set of copyrightable works thus published on the MMC site.

"CC-BY-SA" means the Creative Commons Attribution-Share Alike 3.0 license published by Creative Commons Corporation, a not-for-profit corporation with a principal place of business in San Francisco, California, as well as future copyleft versions of that license published by that same organization.

"Incorporate" means to publish or republish a Document, in whole or in part, as part of another Document.

An MMC is "eligible for relicensing" if it is licensed under this License, and if all works that were first published under this License somewhere other than this MMC, and subsequently incorporated in whole or in part into the MMC, (1) had no cover texts or invariant sections, and (2) were thus incorporated prior to November 1, 2008.

The operator of an MMC Site may republish an MMC contained in the site under CC-BY-SA on the same site at any time before August 1, 2009, provided the MMC is eligible for relicensing.

ADDENDUM: How to use this License for your documents

To use this License in a document you have written, include a copy of the License in the document and put the following copyright and license notices just after the title page:

> Copyright (c) YEAR YOUR NAME.
> Permission is granted to copy, distribute and/or modify this document under the terms of the GNU Free Documentation License, Version 1.3 or any later version published by the Free Software Foundation; with no Invariant Sections, no Front-Cover Texts, and no Back-Cover Texts.

A copy of the license is included in the section entitled "GNU Free Documentation License".

If you have Invariant Sections, Front-Cover Texts and Back-Cover Texts, replace the "with...Texts." line with this:

with the Invariant Sections being LIST THEIR TITLES, with the Front-Cover Texts being LIST, and with the Back-Cover Texts being LIST.

If you have Invariant Sections without Cover Texts, or some other combination of the three, merge those two alternatives to suit the situation.

If your document contains nontrivial examples of program code, we recommend releasing these examples in parallel under your choice of free software license, such as the GNU General Public License, to permit their use in free software.