Using the GEOS-5 AGCM

Carlos Cruz

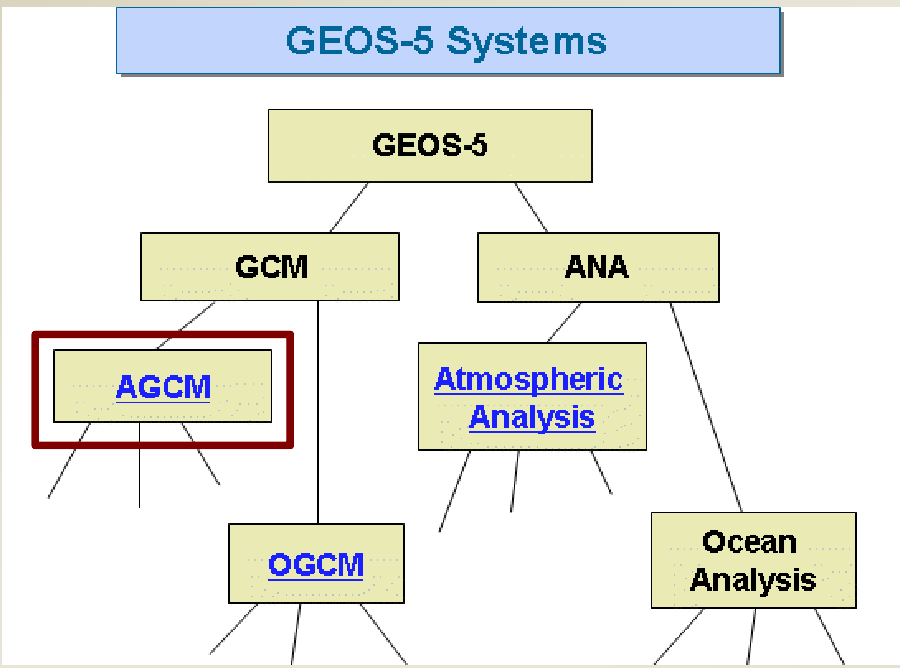
Climate In a Box Workshop

September 21-22, 2010
Goddard Space Flight Center
Greenbelt, MD

Outline

- Introduction to the GEOS-5 AGCM
- Obtaining the code
    - Code versions
- Compilation and installation
    - Code structure
- Experimental setup
    - Setting up GEOS5 on the CIB platform
- Hands-on

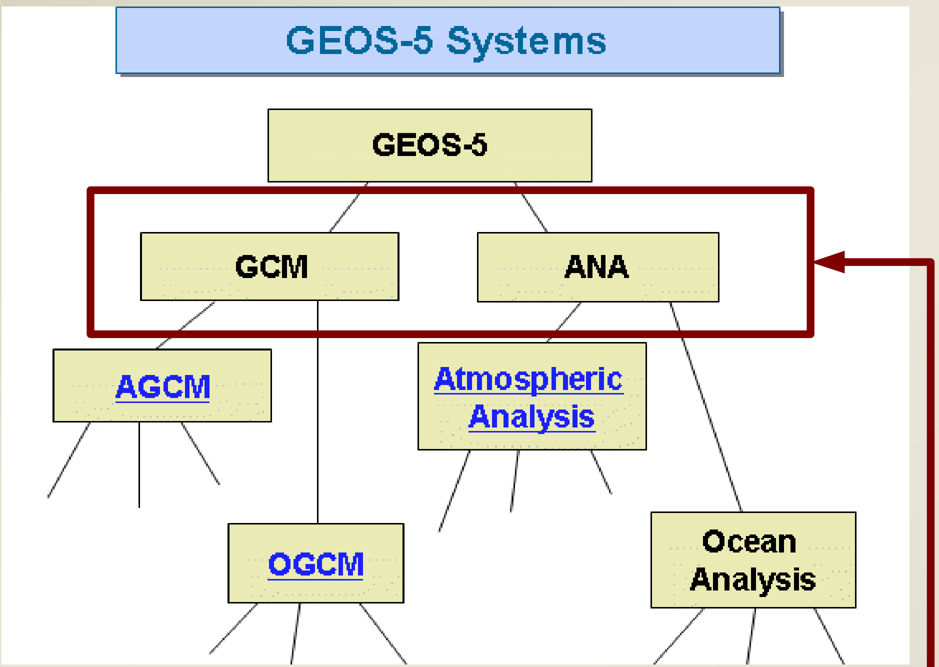
Introduction to the AGCM



The Atmospheric General Circulation Model (AGCM) is part of GEOS-5 and is maintained at the Global Modeling and Assimilation Office (GMAO).

The AGCM is used for atmospheric analyses, weather forecasts, uncoupled and coupled climate simulations and predictions, and for coupled chemistry-climate simulations.

Introduction to the AGCM

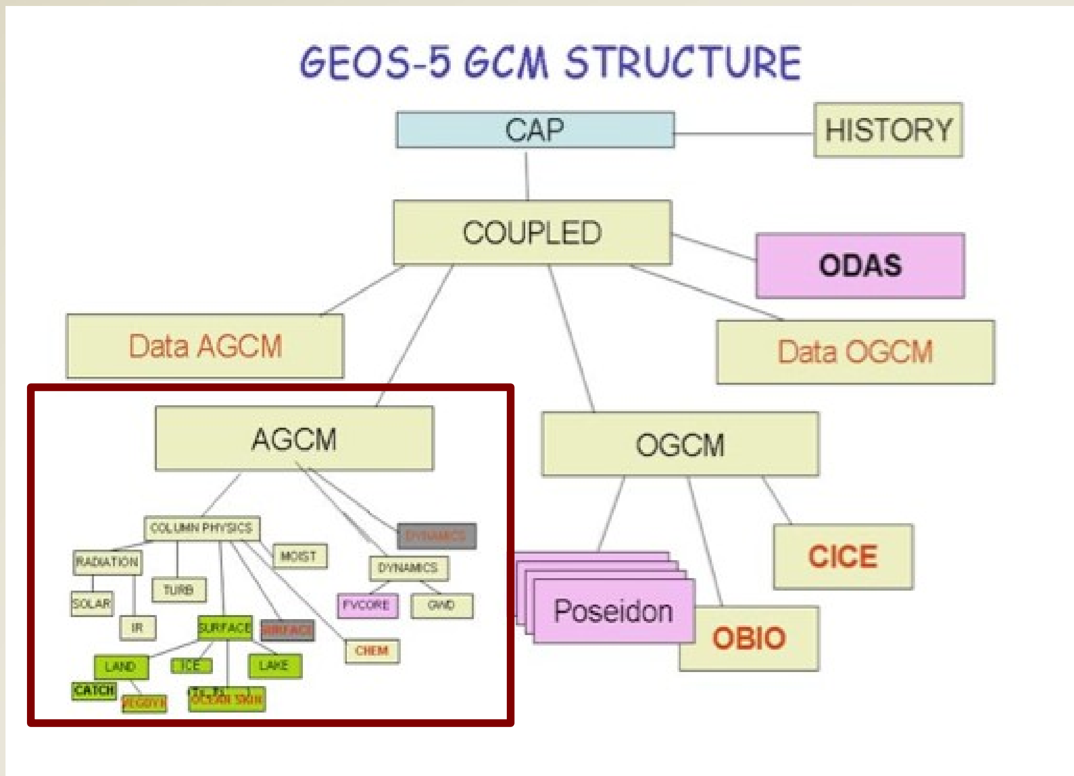


The Atmospheric General Circulation Model (AGCM) is part of GEOS-5 and is maintained at the Global Modeling and Assimilation Office (GMAO).

The AGCM is used for atmospheric analyses, weather forecasts, uncoupled and coupled climate simulations and predictions, and for coupled chemistry-climate simulations.

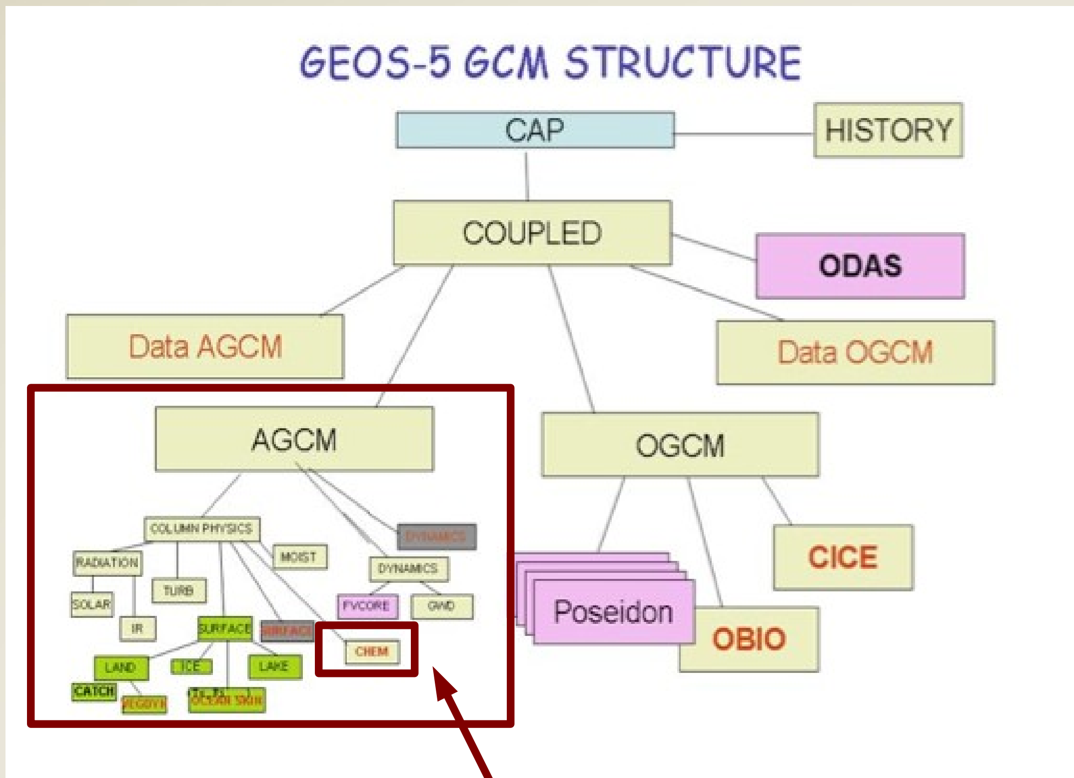
Data Assimilation System (not covered)

GEOS5 code structure



AGCM

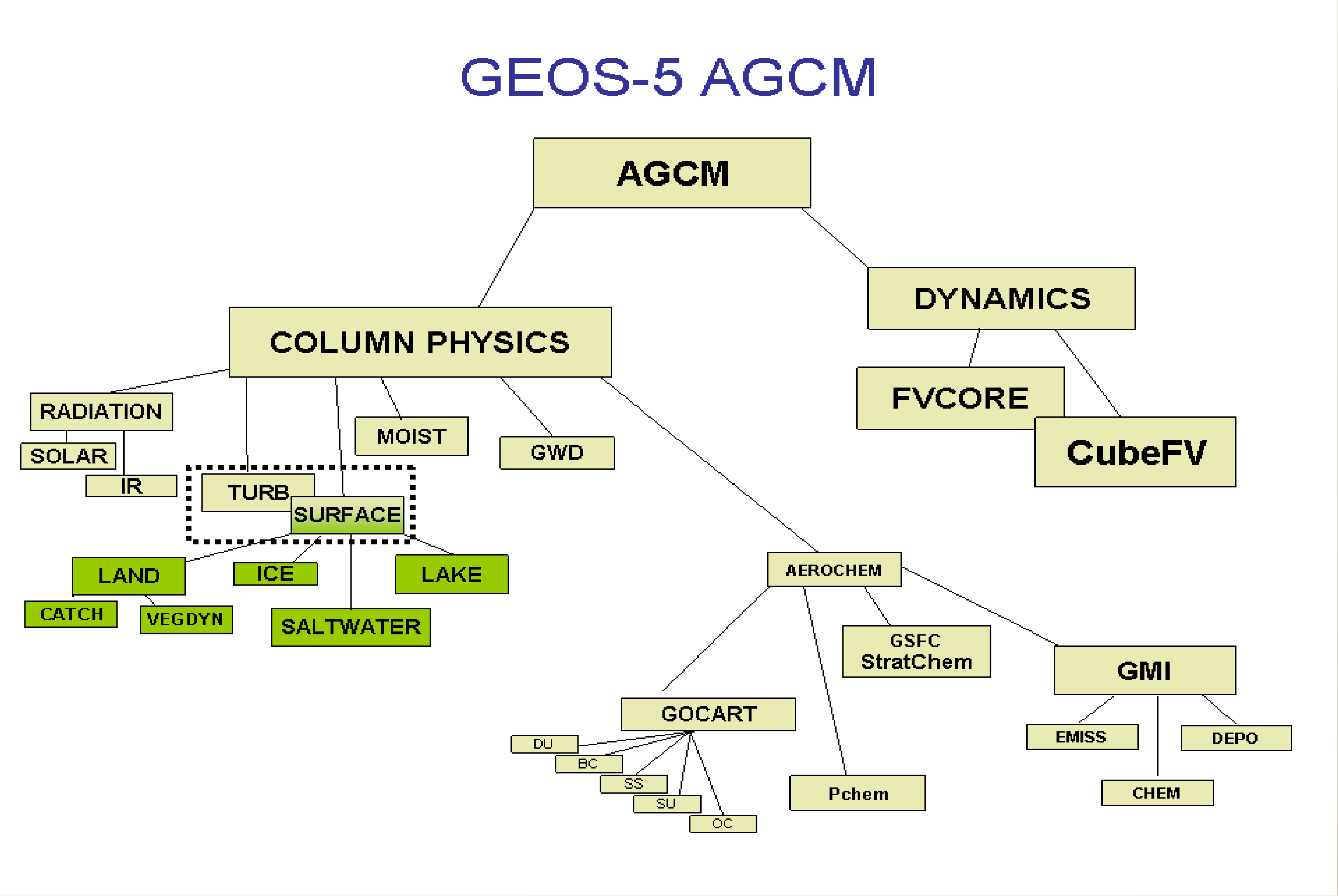
GEOS5 code structure



AGCM

Each box is an ESMF component
ESMF: Earth System Modeling Framework

AGCM code structure



Obtaining the code

- Categories of users:

  (1) Those who will use the model as a “black box”

  (2) Those who will modify the code.

Obtaining the code

- On NCCS systems please refer to the GEOS5 user's guide:
  - http://modelingguru.nasa.gov/docs/DOC-1393
- On non-NCCS systems, you must first get an NCCS account
  - http://www.nccs.nasa.gov
- Then an account on the CVS repository, “progress”:
  - https://progress.nccs.nasa.gov/trac/admin/wiki/QuickStart
  - and/or refer to CIB user's guide

Obtaining the code

- Code[1] is already installed and ready to run on the CIB system!

  /cib/models/geos5gcm/GEOSagcm-Eros_7_25

  /cib/models/geos5gcm/Fortuna-2.1[2]

- With corresponding input data (BCs, restarts) on

  /cib/inputdata/geos5gcm/MERRA

  /cib/inputdata/geos5gcm/Fortuna-2.1[2]

[1] Two CVS tags: GEOSagcm-Eros_7_25 and Fortuna-2.1

[2] When ported.

Code versions

- GEOS-5 AGCM revision history
  http://modelingguru.nasa.gov/docs/DOC-1411
- GEOSagcm-Eros_7_25 → MERRA tag
  - FV (lat-lon) dynamical core, serial IO

↑

Modern Era Retrospective-analysis for Research and Applications

Code versions

- GEOS-5 AGCM revision history
  http://modelingguru.nasa.gov/docs/DOC-1411
- GEOSagcm-Eros_7_25 ⟶ MERRA tag
  - FV (lat-lon) dynamical core, serial IO
- Fortuna tags (e.g. 1.4, 2.1)
  - Write output collections in parallel
  - MAPL upgraded to ESMF 3.x or 4.x
  - Can run with up to 4 dynamical cores (2.x)
    - FV (lat-lon)
    - FV (cubed-sphere)
    - ARIES (lat-lon)
    - DATMO (data model for single column physics)

Baselibs

- Baselibs are a set of support libraries (ESMF, HDF, netCDF, etc) that need to be installed before attempting to build the GEOS-5 AGCM application.

  cvs -d :ext:<uid>@<repository_address>:/cvsroot/baselibs co -r SOMETAG Baselibs

  case sensitive

- Some SOMETAG examples:
  - GMAO-Baselibs-2_1_1 (MERRA)
  - GMAO-Baselibs-3_1_4 (Fortuna 1.4)
  - GMAO-Baselibs-3_2_0 (Fortuna 2.1)
- Baselibs are already installed and ready to be used on the CIB system.

https://modelingguru.nasa.gov/docs/DOC-1685

Obtaining the code

- The GEOS-5 (AGCM) source code checks out at about 44/184 MB of space for MERRA/Fortuna tags respectively.

  ```
  cvs -d $CVSROOT checkout -d DIRECTORY -r TAGNAME MODULENAME
  ```

- Here, $CVSROOT specifies the CVS repository we'll be getting the code from
- DIRECTORY is the name of the directory you would like to create to hold the code
- MODULENAME is the particular module (set of code) we'll be checking out, and
- TAGNAME is a particular version of that module.

Obtaining the code

- For the AGCM

cvs -d $CVSROOT co -r GEOSagcm-Eros_7_25 GEOSagcm

cvs -d $CVSROOT co -r Fortuna-2_1 Fortuna

Where $CVSROOT is :ext:<uid>@progress.nccs.nasa.gov:/cvsroot/esma

Obtaining the code

- For the AGCM

cvs -d $CVSROOT co -r GEOSagcm-Eros_7_25 GEOSagcm

cvs -d $CVSROOT co -r Fortuna-2_1 Fortuna

Note module name is the Same as the tag

Where $CVSROOT is :ext:<uid>@progress.nccs.nasa.gov:/cvsroot/esma

- If you want to avoid typing in your password all the time (recommended) you need to set up your ssh keys.
  - See for example

    http://modelingguru.nasa.gov/docs/DOC-1112

Obtaining the code

- What do you get:

  $ *cvs -d $CVSROOT co -r GEOSagcm-Eros_7_25 GEOSagcm*

  Upon checkout you will get GEOSagcm directory

- $ *cd GEOSagcm/src* ← Source directory

  - Applications
  - Assert.pl
  - Config
  - CVS
  - g5_modules
  - GEOSgcs_GridComp
  - GMAO_Shared
  - GNUmakefile
  - parallel_build.csh

Compilation and Installation

- So, the first thing to do is to set up your environment to use the appropriate compilers and libraries.

Modules

- The **module** package allows dynamic modification of a user's programming environment via module files.
- The **module** command can be used to:
  - Manage multiple versions of compilers, applications, tools and libraries
  - Manage software where complex changes to the environment are necessary
  - Manage software where name conflicts with other software would cause problems
- The **module** package is available automatically for interactive and batch use on the CIB system.

Compilation and Installation

- On the CIB system there's a helper script to help set up the module environment:

  cp /cib/libraries/archives/set_mode.bash ~/

- To use:

  . *set_mode* <module_set>

  - Where <module_set> is a short string that specifies a particular module environment
  - Examples of <module_set> : Fort21, MerraI9

Compilation and Installation

- There's one other environment variable to (optionally) set:

  export *ESMADIR*=<install_path>/GEOSagcm

- This sets the location of where the build gets installed. It should default to this location, but in case you have multiple instances of the code checked out it's a good idea to be explicit.
- Now, assuming you're in the source directory, you can build the model executable issuing the following command:

  gmake install &> makefile.log &

- or

  gmake install FOPT=-g &> makefile.log &

FOPT allows for compile options
To be set at the command line

Building the code

- So, *gmake install* builds the executable
- But note there are dependencies
  - Generated during the build process (.d files)
  - *May* reduce recompilation times after changes
- Errors/Warnings
  - Most are harmless, else if there is a fatal error the compilation will fail
- To speed up the compilation (45 mins) you may want to use parallel_build.csh.

Code structure

- If installation is successful, step back up out of the source directory and you should see the following sub-directories:

  - Config
  - CVS
  - Linux
    - bin ← Executables are here. Look for GEOSgcm.x
    - Config
    - doc
    - etc ← rc and tmpl files
    - include
    - lib
  - src

Break

Experimental Setup

- Once model has been installed, some ENV variables may be helpful

  setenv GEOSUTIL <install_path>/GEOSagcm/src/GMAO_shared/GEOS_Util

  setenv PATH .:<install_path>/GEOSagcm/Linux/bin:$PATH

  GEOSUTIL contains various utilities used to manipulate diagnostics.

- Now we need to create an experiment directory (scratch) that will contain GEOSgcm.x, restarts, BCs and resource files.
- Finally we need a run script.

  On a Linux cluster managed by a job queueing system the run script sets up all ENV variables, module ENV, creates links to data directories and executes the model. Additionally it may perform post-processing and data archival.

- gcm_setup under src/Applications/GEOSgcm_App

Experimental Setup

- Once model has been installed, some ENV variables may be helpful

  setenv GEOSUTIL <install_path>/GEOSagcm/src/GMAO_shared/GEOS_Util

  setenv PATH .:<install_path>/GEOSagcm/Linux/bin:$PATH

  GEOSUTIL contains various utilities used to manipulate diagnostics.

- Now we need to create an experiment directory (scratch) that will contain GEOSgcm.x, restarts, BCs and resource files.
- Finally we need a run script.

  On a Linux cluster managed by a job queueing system the run script sets up all ENV variables, module ENV, creates links to data directories and executes the model. Additionally it may perform post-processing and data archival.

- gcm_setup ← Does not work on CIB platform but it can be adapted.

Experimental Setup

- On CIB we have a tool to help expedite an experiment setup:

  Located in /cib/models/archive/AGCMsetup.tgz

  Setup MERRA or Fortuna run (2^{o} resolution only).

- Basically

  cp /cib/models/archive/AGCMsetup.tgz /cib/outputdata/<userid>

  tar xfz AGCMsetup.tgz

  cd AGCMsetup

  - EDIT options.rc file
  - RUN setupg5gcm
  - This will be demonstrated in a few minutes.

Experimental Setup

- AGCMsetup helps setup MERRA or Fortuna run (2^{o} resolution only).
- You can keep the setup to continue with code development

- We still need to create a run script to run the model.
- Note that one can run model interactively
  - qsub -I -l nodes=2:ppn=8
  - cd /cib/outputdata/<USERNAME>/<EXPERIMENT_DIRECTORY>
- On CIB, use /cib/models/archives/merra_run.j
  - Minor edits need to be made.

CIB experiment setup

- Use CIB GEOS5 AGCM experiment setup tool.

  cd /cib/outputdata/ccruz2

  cp /cib/models/archive/AGCMsetup.tgz ./

  tar xfz AGCMsetup.tgz

  cd AGCMsetup

  vi options.rc

  - Edit as needed

  setupg5gcm

Please do not attempt these steps...yet.

RC files

- CAP.rc

```
MAPLROOT_COMPNAME: GCS
        ROOT_NAME: GCS
ROOT_CF: AGCM.rc
HIST_CF: HISTORY.rc

NX: 4
NY: 4
IM: 144
JM: 91
LM: 72

BEG_DATE:      20070408 210000
END_DATE:      20070413 210000
JOB_DURATION: 0000005 000000
RUN_DT:        1800

MAPL_ENABLE_TIMERS: NO
```

Domain decomposition NX*NY=NCPUS

RC files

- CAP.rc

```
MAPLROOT_COMPNAME: GCS
        ROOT_NAME: GCS
ROOT_CF: AGCM.rc
HIST_CF: HISTORY.rc

NX: 4
NY: 4
IM: 144
JM: 91
LM: 72

BEG_DATE:     20070408 210000
END_DATE:     20070413 210000
JOB_DURATION: 0000005 000000
RUN_DT:       1800

MAPL_ENABLE_TIMERS: NO
```

Resolution

RC files

- CAP.rc

```
MAPLROOT_COMPNAME: GCS
        ROOT_NAME: GCS
ROOT_CF: AGCM.rc
HIST_CF: HISTORY.rc

NX: 4
NY: 4
IM: 144
JM: 91
LM: 72

BEG_DATE:      20070408 210000
END_DATE:      20070413 210000
JOB_DURATION: 0000005 000000
RUN_DT:        1800

MAPL_ENABLE_TIMERS: NO
```

Ignore BEG_DATE: Your experiment directory should contain a file called cap_restart that will have the model start date

RC files

- AGCM.rc

```
# Model Resolution and Timestep Parameters
# -----------------------------------------
NX:      4
NY:      4

AGCM_GRIDNAME: PC144x91-DC
OGCM_GRIDNAME: PE360x180-DE

AGCM_IM:      144
AGCM_JM:      91
AGCM_LM:      72

OGCM_IM: 360
OGCM_JM: 180
OGCM_LM: 34

  RUN_DT: 1800
SOLAR_DT: 3600
IRRAD_DT: 3600

SOLARAvrg: 0
IRRADAvrg: 0
```

Domain decomposition NX*NY=NCPUS

Resolution

RC files

- AGCM.rc

Restart file name

```
# Restart and Checkpoint Files
# ----------------------------
DYN_INTERNAL_RESTART_FILE:          fvcore_internal_restart
DYN_INTERNAL_CHECKPOINT_FILE:       fvcore_internal_checkpoint
DYN_INTERNAL_HEADER:                1

LAKE_INTERNAL_RESTART_FILE:         lake_internal_restart
LAKE_INTERNAL_RESTART_TYPE:         binary
LAKE_INTERNAL_CHECKPOINT_FILE:      lake_internal_checkpoint
LAKE_INTERNAL_CHECKPOINT_TYPE:      binary

LANDICE_INTERNAL_RESTART_FILE:      landice_internal_restart
LANDICE_INTERNAL_RESTART_TYPE:      binary
LANDICE_INTERNAL_CHECKPOINT_FILE:   landice_internal_checkpoint
LANDICE_INTERNAL_CHECKPOINT_TYPE:   binary

etc...
```

Add # to
Comment out

Checkpoint file name

RC files

- AGCM.rc

BCs →

```
TOPO_MEAN_FILE:     topo_dynave.data
TOPO_GWDVAR_FILE:   topo_gwdvar.data
TOPO_TRBVAR_FILE:   topo_trbvar.data

laigrn_clim:             laigrn.data
TILING_FILE:               tile.data
VISDF_FILE:                visdf.dat
NIRDF_FILE:                nirdf.dat
```

Aerosols/ chemistry parameters →

```
# AeroChem Environment
# --------------------
    CHEM_METHOD: PCHEM
   OX_RELAXTIME: 0.001
  CH4_RELAXTIME: 0.001
  N2O_RELAXTIME: 0.001
CFC11_RELAXTIME: 0.001
CFC12_RELAXTIME: 0.001
CFC22_RELAXTIME: 0.001
  H2O_RELAXTIME: 259200.
  OX_FRIENDLIES: ANALYSIS
#AOA_FRIENDLIES: DYNAMICS:TURBULENCE
     pchem_clim: species.data

AEROCLIM:    ExtData/g5chem/L72/aero_clm/gfedv2.aero.eta.%y4%m2clm.hdf
AEROCLIMDEL: ExtData/g5chem/L72/aero_clm/gfedv2.del_aero.eta.%y4%m2clm.hdf
AEROCLIMYEAR: 2002
```

BCs →

```
DU_OPTICS: ExtData/g5chem/x/opticsBands_DU.hdf
SS_OPTICS: ExtData/g5chem/x/opticsBands_SS.hdf
SU_OPTICS: ExtData/g5chem/x/opticsBands_SU.hdf
OC_OPTICS: ExtData/g5chem/x/opticsBands_OC.hdf
BC_OPTICS: ExtData/g5chem/x/opticsBands_BC.hdf
NUM_BANDS: 18
```

RC files

- HISTORY.rc

```
EXPID:  geosagcm
EXPDSC: geosagcm_GEOSagcm-Eros_7_25

COLLECTIONS: 'geosgcm_prog'
             'geosgcm_surf'
             ::

  geosgcm_prog.template:  '%y4%m2%d2_%h2%n2z.hdf',
  geosgcm_prog.format:    'CFIO',
  geosgcm_prog.frequency:  060000,
  geosgcm_prog.duration:   240000,
  geosgcm_prog.vscale:     100.0,
  geosgcm_prog.vunit:     'hPa',
  geosgcm_prog.vvars:     'log(PLE)' , 'DYN'          ,
  geosgcm_prog.levels:     1000 975 950 925 900 875 850 825 800 775 750 725 700 650 600 550 500 450 400
 350 300 250 200 150 100 70 50 40 30 20 10 7 5 4 3 2 1 0.7 0.5 0.4 0.3 0.2 0.1 0.07 0.05 0.04 0.03 0.02
,
  geosgcm_prog.fields:    'PHIS'     , 'SUPERDYNAMICS',
                          'SLP'      , 'DYN'          ,
                          'U'        , 'DYN'          ,
                          'V'        , 'DYN'          ,
                          'T'        , 'DYN'          ,
                          'PS'       , 'DYN'          ,
                          'ZLE'      , 'DYN'          , 'H'    ,
                          'OMEGA'    , 'DYN'          ,
                          'Q'        , 'MOIST'        , 'QV'   ,
                          'QITOT'    , 'MOIST'        , 'QI'   ,
                          'QLTOT'    , 'MOIST'        , 'QL'   ,
                          'RH2'      , 'MOIST'        , 'RH'   ,
                          'O3'       , 'CHEMISTRY'    ,
                          ::

  geosgcm_surf.template:  '%y4%m2%d2.hdf',
  geosgcm_surf.format:    'CFIO',
  geosgcm_surf.mode:      'time-averaged',
  geosgcm_surf.frequency:  240000,
  geosgcm_surf.duration:   240000,
  geosgcm_surf.ref_time:   210000,
  geosgcm_surf.fields:    'PHIS'     , 'SUPERDYNAMICS' ,
                          'SGH'      , 'SUPERDYNAMICS' ,
                          'VARFLT'   , 'SUPERDYNAMICS' ,
                          'TROPT'    , 'AGCM'          ,
```

Additional info: http://modelingguru.nasa.gov/docs/DOC-1190

Model Output

- All GEOS-5 output streams are GrADS readable (http://grads.iges.org/).
- MERRA: HDF and GrADS binary.
- Fortuna: netCDF, HDF and GrADS binary format.
- There are some built-in utilities for generating monthly-mean plots in GEOS_Util using GrADS

CIB hands-on

- To do on your laptop: Log on, open up a terminal window (X term)

  ssh nimbus

  cd /cib/outputdata/guest{X} where X=1...9

  cp /cib/inputdata/archives/GEOSagcm-Eros_7_25.scratch.tar .

Note the space followed by a dot

GEOSagcm-Eros_7_25.scratch.tar contains data, rc files and model executable necessary to run a 2° resolution simulation. This is what AGCMsetup produces.

CIB hands-on

- To do on your laptop: Log on, open up a terminal window (X term)

ssh nimbus

cd /cib/outputdata/guest{X} where X=1…9

cp /cib/inputdata/archives/GEOSagcm-Eros_7_25.scratch.tar .

tar xf GEOSagcm-Eros_7_25.scratch.tar

cd GEOSagcm-Eros_7_25.scratch

CIB hands-on

- To do on your laptop: Log on, open up a terminal window (X term)

ssh nimbus

cd /cib/outputdata/guest{X} where X=1...9

cp /cib/inputdata/archives/GEOSagcm-Eros_7_25.scratch.tar .

tar xf GEOSagcm-Eros_7_25.scratch.tar

cd GEOSagcm-Eros_7_25.scratch

cp /cib/models/archives/merra_env.bash .

. merra_env.bash

↑

Note the dot followed by a space

CIB hands-on

- At this point we have setup the module environment appropriate for the GEOS5 AGCM MERRA tag. Now we need a run script:

Note the space followed by a dot

cp /cib/models/archives/merra_run.j .

Open up *merra_run.j* with your favorite editor

CIB hands-on

- At this point we have setup the module environment appropriate for the GEOS5 AGCM MERRA tag. Now we need a run script:

```
#!/bin/bash
#PBS -S /bin/bash
#PBS -N geos5
#PBS -l nodes=2:ppn=8
#PBS -l walltime=1:00:00
#PBS -j eo

cd /cib/outputdata/guestX/GEOSagcm-Eros_7_25.scratch

# set module environment
. ./merra_env.bash

mpirun_rsh -hostfile $PBS_NODEFILE -np 16 GEOSgcm.x
```

change X to number from 1...9

CIB hands-on

- When done **submit** the "job" to the batch system. To **submit** the job we use the "qsub" command:

  qsub merra_run.j

- When can monitor the job via the "qstat" command:

  qstat

- *You will get something like this:*

  Job id Name User Time Use S Queue

  ------------------- ---------------- --------------- -------- - -----

  266.nimbus geos5 guest2 00:00:32 R workq

CIB hands-on

- When run completes, under "scratch" directory you will see three files with **.hdf** extension:

  geosagcm.geosgcm_prog.20070409_0300z.hdf

  geosagcm.geosgcm_prog.20070409_2100z.hdf

  geosagcm.geosgcm_surf.20070409.hdf

- To visualize model output we use GrADS.
- Run as follows

Note the space followed by a dot

cp /cib/models/archives/merra_view.gs .

grads -l -c 'run merra_view.gs'

Single quotes

Questions?

Extra slides

Some Useful Module Commands

(*see module man pages for more information*)

module avail	Lists all available modules on a system
module list	Lists modules currently in your environment
module load *foo*	Loads module "*foo*"
module unload *foo*	Undo all changes to the environment made by previously loading module "*foo*"
module purge	Unloads all loaded module files
module switch *foo bar*	Switches between module "*foo*" and module "*bar*"
module display *foo*	Indicates what changes would be made to the environment by loading module "*foo*" without actually loading it
module help *foo*	Displays help on module "*foo*"
module whatis *foo*	Displays a brief description of module "*foo*"

Reinitializing Modules

- If you switch shell from your login shell during an “interactive” or “interactive batch” session, you need to reinitialize “module” before you can use it again.
- If you specify a shell in your batch job that is different from your login shell, you need to reinitialize “module” before you can use it again from within the batch script.
- To reinitialize “module”:
  - For csh type shells (csh, tcsh)

    `source  /opt/modules/Modules/3.2.5/init/`*`shell-name`*

    where *`shell-name`* is either `csh` or `tcsh`.

  - For sh type shells (sh, bash, ksh)

    /opt/modules/Modules/3.2.5/init/*`shell-name`*

    where *`shell-name`* is one of `sh`, `bash`, or `ksh`.