LMDZ tutorial

Configurations, input files, forcing data

LMDZ tutorial
December 2016
Lionel GUEZ
Laboratoire de météorologie dynamique

Contents

- Configurations
- LMDZ with realistic physics and surface
- Aqua-planet or all-land planet
- Relaxation to a 3D-temperature field

Configurations: many ways to use LMDZ

Coupling with another model

- Soil, vegetation, hydrology: Orchidée
http://labex.ipsl.fr/orchidee
- Ocean, sea ice, marine biogeochemistry: Nemo
http://www.nemo-ocean.eu
- Tropospheric chemistry, aerosols: INCA
http://www-lsceinca.cea.fr
- Stratospheric chemistry: Reprobus
Marchand et al., JASTP, 2012

Configurations of LMDZ itself

- 1 or 3 dimensions
- Nudging or not
- Zoom or not
- Realistic or idealized ”physical” part of the model

Use case:
LMDZ alone
3 dimensions
no nudging
realistic (terrestrial) physics and surface

What is left to choose

- Some run-time parameters (among which concentration of greenhouse gases)
- Initial state
- Boundary (that is, surface) conditions
- Forcing atmospheric data: ozone field, fields of aerosols

Run-time parameters (1/6)

- Chosen in text files which have suffix `.def`
- Home › Utilisateurs › Guides › LMDZ pas à pas › Utilisation comme boîte noire › Variables des fichiers .def

Run-time parameters (2/6)

- Syntax:
  `variable=value`
  Case sensitive
  TRUE or FALSE for logical values
  Character values without quotes
  Comment lines start with #
  Example: gcm.def
- Semantic separation of run-time parameters among seven `.def` files

Run-time parameters (3/6)

- `gcm.def`: dynamics
  - Zoom: clon and clat (center), grossism[xy] (zooming factor), dzoom[xy] (range), tau[xy] (steepness of transition)
- `physiq.def`: parameterizations
- `output.def`: fine tuning of output (choice of variables, frequency...)
- vert.def: vertical discretization

Run-time parameters (4/6)

- `config.def`: level of output, coupling with other models, radiative transfer, forcing data (ozone, aerosols, greenhouse gases)
  - No ocean model: type_ocean=force
  - No coupling with Orchidée: VEGET=bucket or VEGET=betaclim
  - Concentration of greenhouse gases: co2_ppm, ch4_ppb...
  - Number of calls to radiative transfer procedure, per day: nbapp_rad

Run-time parameters (5/6)

- `run.def`: length of the run, dates, calendar
  - Number of days of run: nday
  - Starting date of the first run in a sequence of runs: dayref (day number in the year) and anneeref. You do not change these when you continue a simulation, starting from the end of a previous simulation.
  - Calendar: calend (character variable)

Run-time parameters (6/6)

- traceur.def: tracers
  - Special syntax
  - 1st line: number of tracers
  - Then one line per tracer, with a number identifying the advection scheme
  - For more information on available advection schemes: dyn3d_common/infotrac.F90

INCLUDEDEF lines in .def files

- `run.def` is the first file read.
- Other `.def` files are read only if there is a corresponding `INCLUDEDEF` line in `run.def`. Cf. distributed `run.def`.

List of all used parameters (1/2)

- Parameters have default values in the Fortran source files.
- →The `.def` files that you will provide will probably not mention all the parameters.
- Also, be careful: you will not get an error message for a mistyped parameter name in your `.def` files, it will just be ignored.

List of all used parameters (2/2)

- So, if you want to see all the parameter values that were used for your run: each run creates a set of files `used_*.def`.
- Script for easier comparison of two sets of `.def` files: http://lmdz.lmd.jussieu.fr/utilisateurs/outils/utilisation-de-lmdz#section-5

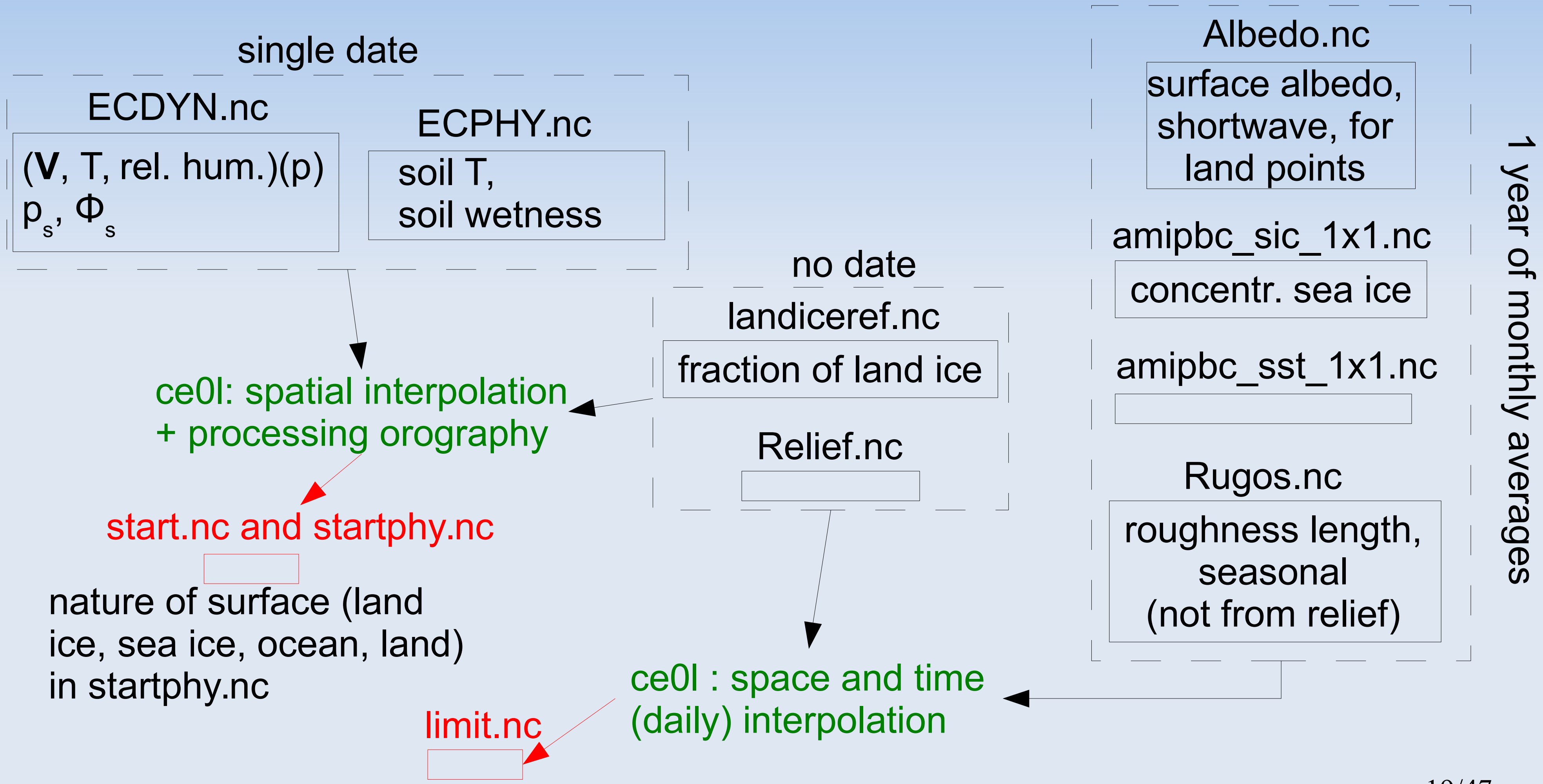
Initial state and boundary conditions (1/3)

- Separate program: ce0l
  (ce0l stands for ”création état 0 limites”)
- Distributed with the program gcm, compiled with the same tools
- Run ce0l with the same `.def` files than those you want for gcm.

Initial state and boundary conditions (2/3)

- ce0l creates 2 files for initial state:
  start.nc for the dynamics part of gcm
  startphy.nc for the physics part of gcm
- ce0l creates 1 file for 1 year of boundary conditions:
  limit.nc

Initial state and boundary conditions (3/3)



single date

ECDYN.nc

(**V**, T, rel. hum.)(p)
p_s, Φ_s

ECPHY.nc

soil T,
soil wetness

no date

landiceref.nc

fraction of land ice

Relief.nc

Albedo.nc

surface albedo, shortwave, for land points

amipbc_sic_1x1.nc

concentr. sea ice

amipbc_sst_1x1.nc

Rugos.nc

roughness length, seasonal (not from relief)

1 year of monthly averages

ce0l: spatial interpolation
+ processing orography

start.nc and startphy.nc

nature of surface (land ice, sea ice, ocean, land) in startphy.nc

ce0l : space and time (daily) interpolation

limit.nc

Calendar (1/2)

- Calendar is used in ce0l to create boundary conditions
- For years with constant length, 360 or 365 days:
  calend=earth_360d
  or
  calend=earth_365d

Calendar (2/2)

- For the real calendar (with leap-years), run ce0l for each year you want boundary conditions for:
  calend=gregorian
  anneeref=<the right year>

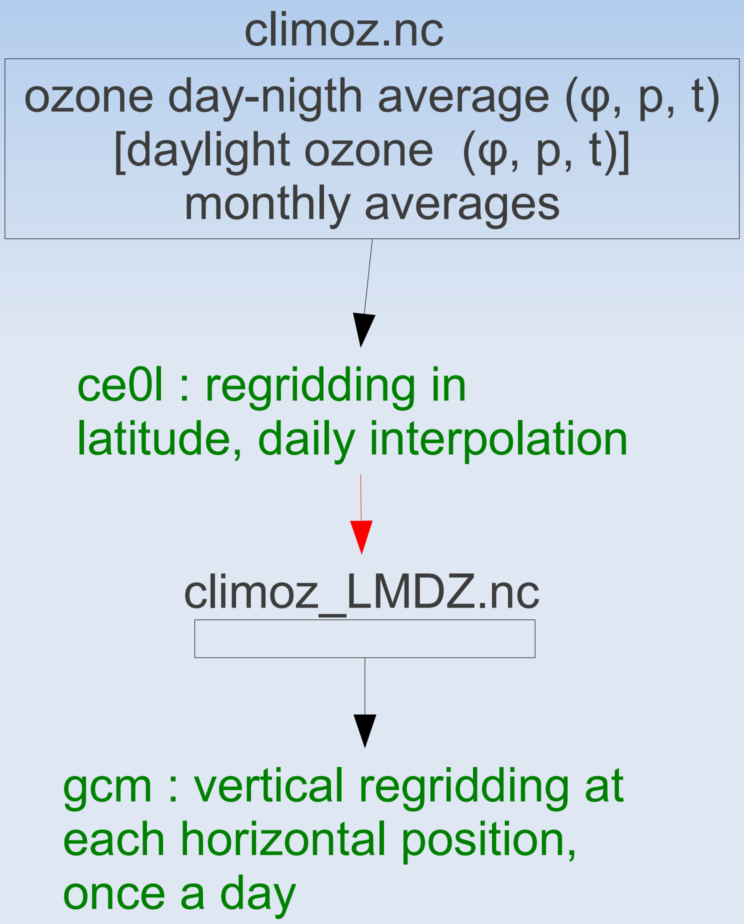
Ozone (1/3)

- Run-time parameter in config.def:
read_climoz = - 1, 0, 1 or 2
- 0: analytic expression for the ozone field (see phylmd/ozonecm_m.F90)
- - 1 (use with solarlong0 = 1000): made symmetric with respect to the equator, for use with annual mean insolation

Ozone (2/3)

- 1: read day-night average of ozone field from a NetCDF file
- 2: also read daylight average from the same file (good idea with Reprobus CMIP5 input and if you run LMDZ with a few model layers in the mesosphere)

Ozone (3/3)



climoz.nc

ozone day-nigth average (φ, p, t)
[daylight ozone (φ, p, t)]
monthly averages

ce0l : regridding in
latitude, daily interpolation

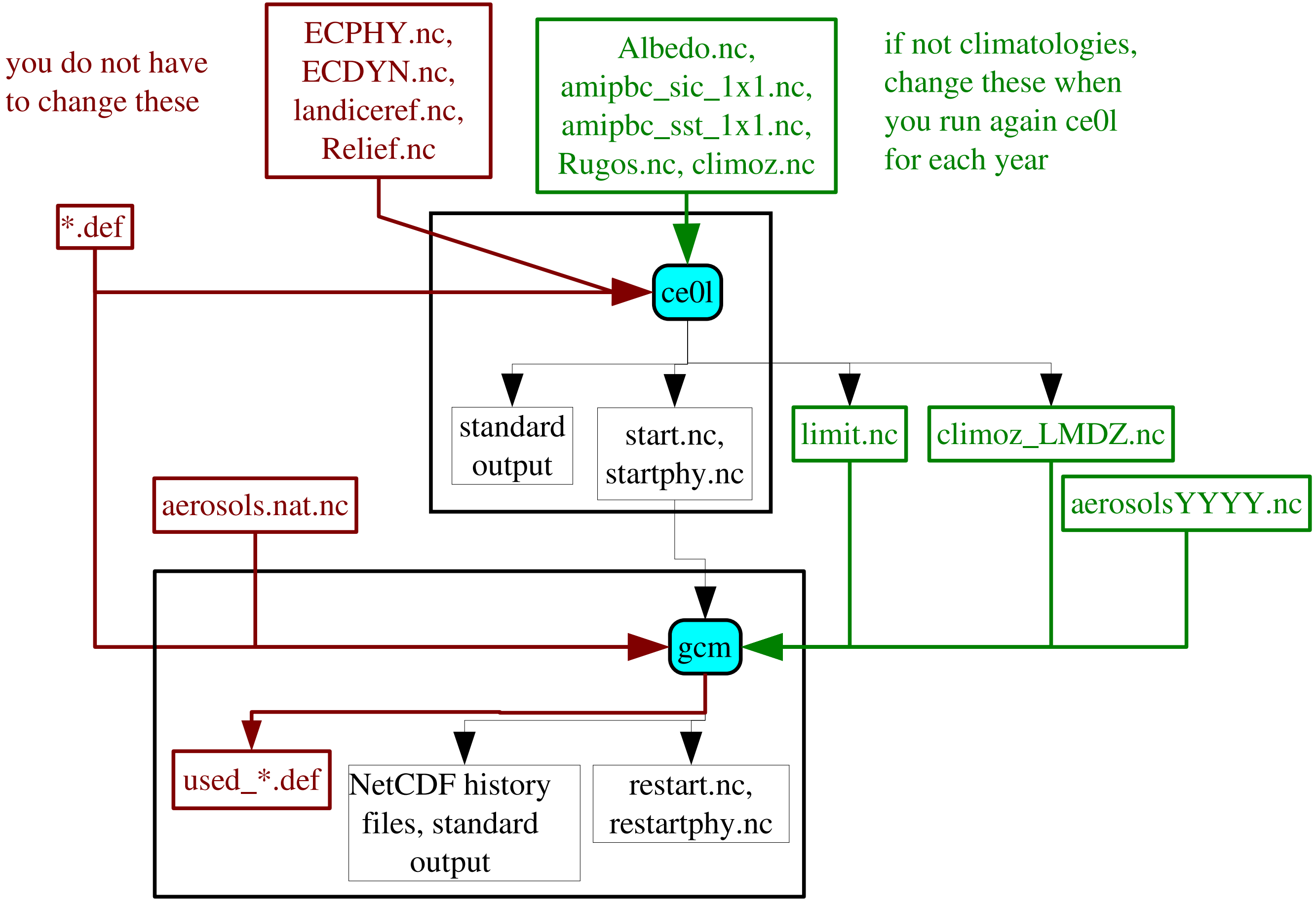
climoz_LMDZ.nc

gcm : vertical regridding at
each horizontal position,
once a day

Aerosols

Cf. presentation by Olivier BOUCHER

Putting it all together



you do not have
to change these

ECPHY.nc,
ECDYN.nc,
landiceref.nc,
Relief.nc

Albedo.nc,
amipbc_sic_1x1.nc,
amipbc_sst_1x1.nc,
Rugos.nc, climoz.nc

if not climatologies,
change these when
you run again ce0l
for each year

*.def

ce0l

standard output

start.nc, startphy.nc

limit.nc

climoz_LMDZ.nc

aerosols.nat.nc

aerosolsYYYY.nc

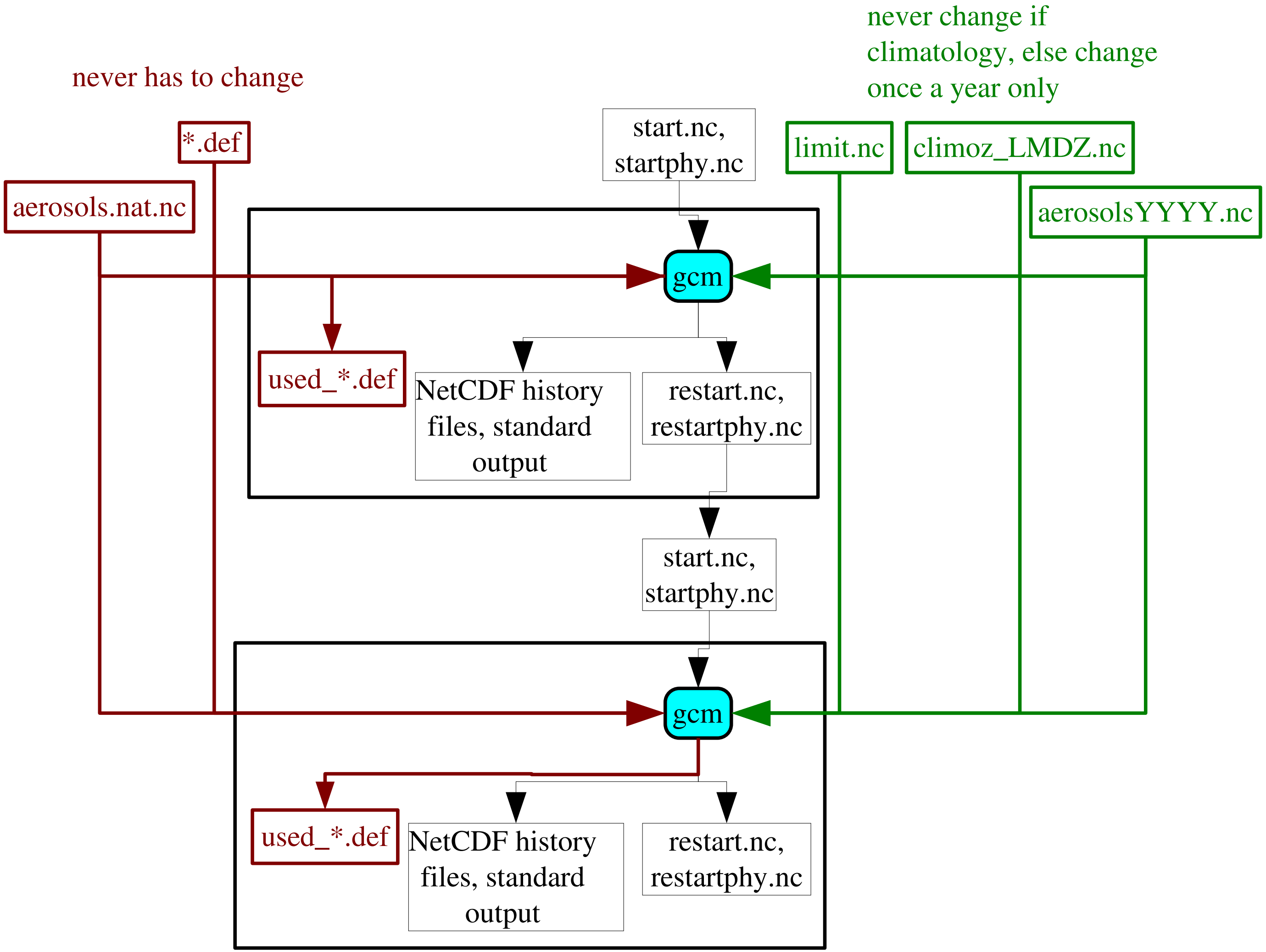
gcm

used_*.def

NetCDF history files, standard output

restart.nc, restartphy.nc

Chaining one-month runs



never has to change
never change if climatology, else change once a year only
*.def
aerosols.nat.nc
start.nc, startphy.nc
limit.nc
climoz_LMDZ.nc
aerosolsYYYY.nc
gcm
used_*.def
NetCDF history files, standard output
restart.nc, restartphy.nc
start.nc, startphy.nc
gcm
used_*.def
NetCDF history files, standard output
restart.nc, restartphy.nc

Where do you get the input files ?

Run-time parameters

Template `.def` files are distributed with LMDZ source files

- Directory DefLists
- physiq.def_L39_AR40.0: 39 levels, physics used for AR4 (old physics)
- physiq.def_L39_NPv3.2: new physics of AR5
- physiq.def_NPv5.67: new physics of AR6 (still evolving)

Initial state and boundary conditions (1/2)

- NetCDF files required by ce0l are at:
  - http://www.lmd.jussieu.fr/~lmdz/LMDZ_Init
  - ergon:~rpsl035/IGCM/INIT/ATM/LMDZ
  - curie:/ccc/store/cont003/dsm/p86ipsl/dmf_import/IGCM/STORAGE/INIT.tar

Initial state and boundary conditions (2/2)

- You will find:
  - landiceref.nc, Relief.nc, Albedo.nc, Rugos.nc
  - sea ice and SST from 1870 to 2015, from AMIP
  - ECDYN.nc and ECPHY.nc for a single date. If you need another date, see: http://lmdz.lmd.jussieu.fr/utilisateurs/outils/utilisation-de-lmdz#section-12

Forcing data for CMIP 5

- A lot of data, for pre-industrial, historical or future simulations (with different scenarios)
- NetCDF files are stored at the national supercomputing centers, IDRIS and TGCC, so you need to find access to them

Forcing data for CMIP 5: ozone (1/2)

- ergon:~rpsl035/IGCM/INIT/ATM/LMDZ/Ozone/HYBRIDE/v2.$scenario/tro3_${year}.new.nc
- For the scenarios:
  - clim: historical then RCP 8.5
  - RCP26, RCP45, RCP60
- curie:/ccc/store/cont003/dsm/p86ipsl/dmf_import/IGCM/STORAGE/INIT.tar

Forcing data for CMIP 5: ozone (2/2)

- Wiki http://forge.ipsl.jussieu.fr/igcmg/wiki/InfosOzone

Forcing data for CMIP 5: aerosols (1/3)

- ergon:~rpsl035/IGCM/ATM/LMD$resolution/AR5/$scenario/aerosols_11YearsClim_${year}_v5.nc

  or

  adapp:/workgpfs/rech/psl/rpsl035/IGCM/ATM/LMD$resolution/AR5/$scenario/aerosols_11YearsClim_${year}_v5.nc

Forcing data for CMIP 5: aerosols (2/3)

- For the resolutions:
  128 118
  128 88
  144 142
  280 280
  96 95

Forcing data for CMIP 5: aerosols (3/3)

- And for the scenarios: HISTORIQUE, RCP26, RCP45, RCP60, RCP85, esm2_experiment
- Each file contains all aerosol types.
- 1855 for aerosols.nat.nc
- Wiki http://forge.ipsl.jussieu.fr/igcmg/wiki/InfosAerosol

Aqua-planet or all-land-planet

Definition

- Zero relief
- Only one type of surface globally : either land or ocean
- Surface temperature is:
  - forced for an aqua-planet
  - computed for a land-planet (we still have to choose an initial field)

Selecting the aqua- or land-planet (1/2)

- Run-time parameter iflag_phys (integer) in gcm.def
  Choose iflag_phys ≥ 100 (instead of default value 1 for Earth surface, full physics)
- iflag_phys = 101 to 107, 109 to 114, 120, 121: aqua-planet
  iflag_phys = 201 to 207, 209 to 214, 220, 221: land-planet

Selecting the aqua- or land-planet (2/2)

- Different T_s fields, constant for aqua-planet, initial value only for land-planet
- See (analytic) definition of the 15 T_s fields in procedure profil_sst (file phylmd/phyaqua_mod.F90)
- Note: all the T_s fields are uniform in longitude and symmetrical with respect to the equator.

Initial state and boundary conditions

- You do not go through the ce0l step
  - Set run-time parameter read_start to FALSE in gcm.def
    gcm creates an initial state
  - gcm also creates a file limit.nc for boundary conditions
- gcm creates restart.nc and restartphy.nc so switch read_start to TRUE for the next run

Forcing data

- For an aquaplanet, it may be a good idea to use adapted sun position, ozone and aerosol fields, symmetrical about the equator
- read_climoz = - 1 and solarlong0 = 1000
- No symmetrical aerosol field ready so either create it yourself or set:
  flag_aerosol = 0

Idealized physics: relaxation to a given 3-dimensional temperature field

Selecting temperature relaxation

Set run-time parameter iflag_phys=2 in gcm.def

- Analytic definition of the 3D equilibrium temperature field
- Damping of low-level wind to represent boundary-layer friction
- From Held and Suarez (1994)
- Some run-time tuning parameters (relaxation time…), see dyn3d/iniacademic.F90

Initial state, no boundary condition

- You do not go through the ce0l step
  - Set run-time parameter read_start to FALSE in gcm.def
    gcm creates an initial state
  - No file limit.nc
- gcm creates restart.nc (no restartphy.nc) so switch read_start to TRUE for the next run

Note

- No other forcing data
- You do not use physiq.def, config.def nor output.def
- You can bypass compilation of physics files with option `-p nophys` of `makelmdz_fcm` → much quicker compilation