# LDMZ tutorial: tracers

LMDZ team

December 9th, 2020

This tutorial focuses on using and adding tracers in LMDZ.
This document can be downloaded as a pdf file (so you could copy/paste command lines from it):

```
wget http://lmdz.lmd.jussieu.fr/pub/Training/Tutorials/Tutorial_Tracers.pdf
```

## 1 Prerequisits

**You should have executed the mandatory part of Tutorial #2.**

## 2 Experimenting with tracers

### 2.1 Radon and lead

Go to the directory TUTORIAL/SIMU1 where you have run a simulation according to Tutorial #2. In the file `traceur.def`, you can see that two trace species, named RN, for radon, and PB, for lead, are already defined in addition to vapor and condensed water. Visualize these two tracers in the file `histday.nc`. The NetCDF variables RN and PB are numbers of atoms per unit mass, in $\mathrm{kg}^{-1}$. Usually, activities are preferred for comparison with observations. The activities per unit mass, in Bq $\mathrm{kg}^{-1}$, are:

$$A_{\mathrm{Rn}} = \frac{\mathrm{RN}}{4{,}765 \cdot 10^5}$$
$$A_{\mathrm{Pb}} = \frac{\mathrm{PB}}{1{,}028 \cdot 10^9}$$

(The numerical values that appear in these fractions are the lifetimes of radon and lead, in s.) You can take a look at LMDZ info number 7, figure 19, or the plot in the General introduction of the LMDZ model (§ 4 "Operating modes") to check that you have sensible results. A recent work on the modeling of tracers with LMDZ is Pilon et al. (2015, QJRMS). If you want to see activities per unit volume, you should divide by the mass density, using NetCDF variables pres, temp and ovap in the file `histhf.nc`. (The Ferret color palette in figure 19 of LMDZ info number 7 is saz2.)

### 2.2 Inserting new tracers

We will now explain how to modify the code in order to add idealized tracers. As an example, we will add two tracers, and call them `NewTr1` and `NewTr2`. We will define a domain of the horizontal grid in which the two tracers will initially have the same constant value. The first tracer will be transported by the boundary layer and convective sub-grid-scale motion, in addition to large-scale advection. The second tracer will only be transported by large-scale advection.

We will have to modify the Fortran program but let us first consider the run-time parameters that we have to modify. At run-time, we decide to include the tracers `NewTr1` and `NewTr2` in the simulation by changing the file `traceur.def`. Change the number of tracers at the first line of `traceur.def` and append one line for each tracer:

```
7
14 14 H2Ov
10 10 H2Ol
10 10 H2Oi
10 10 RN
10 10 PB
10 10 NewTr1
10 10 NewTr2
```

Now let us turn to the Fortran program. The only file we need to modify is

`modipsl/modeles/LMDZ/libf/phylmd/traclmdz_mod.F90`

Here are the changes you should make in that file:

- Declare two new module variables, `id_NewTr1` and `id_NewTr2`, with type integer. These are the identifying numbers of the tracers in the program. You can take a previous declaration (`id_pcsat`, `id_pcocsat` ...) as a template. (Do not forget the OpenMP directives.)
- All remaining changes will be made in the procedure `traclmdz_init`, which is inside the module `traclmdz_mod`. In order to define `id_NewTr1` and `id_NewTr2`, the program will scan the file `traceur.def`, looking for NewTr1 and NewTr2. So you should initialize `id_NewTr1` and `id_NewTr2` to 0 before the loop beginning at line 175. There is a comment just above, saying "Recherche des traceurs connus", which means "looking for known tracers".
- In the body of this loop, set `id_NewTr1` to the value of index `it` if tname(iiq) equals NewTr1. You can add the test near line 265, for instance, after the test for pcq0. You can take another tracer as a template. Do the same for `id_NewTr2`.
- For NewTr2, just after setting `id_NewTr2`, deactivate convective and boundary layer transport by setting `conv_flg(it)` and `pbl_flg(it)` to 0.
- Finally, we will initialize the tracers. There is a loop on tracers, beginning at line 294, which tests whether the initial tracer field, read from the file `start.nc`, is zero everywhere. At this point, the tracer field would also be zero if it was not found in `start.nc`. There is a comment just above the test which says "Initalize tracer that was not found in restart file" (the typo "Initalize" is in the code!). In the body of the test, for our two tracers, change the value at the surface in some horizontal domain. The value of tracers is in the variable `tr_seri`. The first dimension of `tr_seri` is for the horizontal position, the second dimension is for the vertical level and the third dimension identifies the tracer. The index of the surface in the vertical dimension is 1. Use variables `xlat` (latitudes) and `xlon` (longitudes) to choose the horizontal domain.

In summary, after making those changes, svn diff should give you something like this:

```
$ svn diff traclmdz_mod.F90
Index: traclmdz_mod.F90
===================================================================
--- traclmdz_mod.F90 (revision 2304)
+++ traclmdz_mod.F90 (working copy)
@@ -58,6 +58,8 @@
   LOGICAL,SAVE :: rnpb=.FALSE. ! Presence du couple Rn222, Pb210
 !$OMP THREADPRIVATE(rnpb)

+  INTEGER, SAVE:: id_newtr1, id_newtr2
+  !$OMP THREADPRIVATE(id_newtr1, id_newtr2)

 CONTAINS

@@ -172,6 +174,8 @@
 ! ----------------------------------------
     id_rn=0; id_pb=0; id_aga=0; id_be=0; id_o3=0
     id_pcsat=0; id_pcocsat=0; id_pcq=0; id_pcs0=0; id_pcos0=0; id_pcq0=0
+    id_newtr1 = 0
+    id_newtr2 = 0
     DO it=1,nbtr
        iiq=niadv(it+2)
        IF ( tname(iiq) == "RN" ) THEN
@@ -262,6 +266,12 @@
        ELSE IF ( tname(iiq) == "pcq0" .OR. tname(iiq) == "Pcq0" ) THEN
          id_pcq0=it
          conv_flg(it)=0 ! No transport by convection for this tracer
+       else if (tname(iiq) == "NewTr1") then
+          id_newtr1 = it
```

```
+       else if (tname(iiq) == "NewTr2") then
+          id_newtr2 = it
+          conv_flg(it) = 0
+          pbl_flg(it) = 0
        ELSE
           WRITE(lunout,*) 'This is an unknown tracer in LMDZ : ', trim(tname(iiq))
        END IF
@@ -325,6 +335,9 @@
                    tr_seri(i,:,it) = 100.
                 END IF
              END DO
+          else if (it == id_newtr1 .or. it == id_newtr2) then
+             where (xlat >= 40. .and. xlat <= 45. .and. xlon >=0. &
+                  .and. xlon <= 5.) tr_seri(:, 1, it) = 1.
           ELSE
              ! No specific initialization exist for this tracer
              tr_seri(:,:,it) = 0.
```

Now you'll want to recompile the model. In order to do that, go up to `modipsl/modeles/LMDZ` level. Before recompilation :

- rename the old executable `gcm.e` located in `TUTORIAL` (maybe you'll want to use it in another exercise)
- remove the link to it, located in `TUTORIAL/SIMU1` :

```
cd ../..
mv TUTORIAL/gcm.e TUTORIAL/gcm_orig.e
rm TUTORIAL/SIMU1/gcm.e
```

Recompile the model using the `veget=0` option (without the Orchidee model); a new gcm.e file will be produced :

```
./makelmdz -d 48x36x39 -rrtm true gcm
```

Rename the newly created `gcm.e` in `gcm_tracers.e` :

```
mv gcm.e gcm_tracers.e
```

Go again in SIMU1, rename the restart files that you produced when running the model in the frame of Tutorial2, and create a link to the new executable :

```
cd TUTORIAL/SIMU1
mv restart.nc start.nc
mv restartphy.nc startphy.nc
ln -s ../../gcm_tracers.e gcm.e
```

Run the model :

```
./gcm.e > listing
```

Visualize the two new tracers in `histday.nc` and examine the difference between them.