Standalone Example

1. How to produce input *nc files for standalone example

The standalone example utilizes COSMO-ART *nc output files, containing all traced gases within COSMO-ART. In a typical COSMO-ART run not all chemical species are contained in such an *nc file, hence some of them have to be added to the &GRIB_OUT section in INPUT_IO. Only those, that are in a tracer, can be written out. Radicals and fix species are not written out. They have to be added later to the initial concentration vector.

```
!! ** 3D gases inorganic **
'CO       ',
'NO2      ',
'OZONE    ',
'H2O2     ',
'HNO3     ',
'NH3      ',
'NO       ',
'SO2      ',
!! ** 3D gases organic **
'CH4      ',
'HCHO     ',
'ETH      ',
'ISO      ',
!! ** 3D gases radicals **
'HO2      ',
'OH       ',
!! ** All other traced gases **
'SULF     ',
'ALD      ',
'OP1      ',
'OP2      ',
'PAA      ',
'ORA1     ',
'ORA2     ',
'N2O5     ',
'NO3      ',
'PAN      ',
'HC3      ',
'HC5      ',
'HC8      ',
'OL2      ',
'OLT      ',
'OLI      ',
'TOL      ',
'XYL      ',
'ACO3     ',
'TPAN     ',
'HONO     ',
'HNO4     ',
'KET      ',
'GLY      ',
'MGLY     ',
'DCB      ',
'ONIT     ',
'CSL      ',
'API      ',
'LIM      ',
'HACE     ',
'ISHP     ',
'ISON     ',
'MACR     ',
'MAHP     ',
'MPAN     ',
'NALD     ',
'SOA1     ',
'SOA10    ',
'SOA100   ',
'SOA1000  ',
'DMS      ',
'DMSO     ',
```

2. Reading data from *nc file

During the execution of the standalone model, initial concentrations and temperature are read from file. All auxiliary functions are contained in the module **Wrap_NETCDF**.

During the initialization phase within kpp_Main.f90, the function

Get_Boxes(data_kpp, local_temperature, act_hour)

reads the information provided in the netcdf file for time=act_hour for all boxes.

INPUT:
INTEGER :: act_hour: time (in hours) for which to read in the *nc file.

OUTPUT:
REAL:: data_kpp(idim, jdim, kdim, 82)
This array holds one vector of concentrations for every of the (i,j,k) cells. This array is already shuffled to the internal order, that KPP uses. The internal ordering can be reviewed in kpp_Parameters.f90, where indices for each species are denoted.
The array is filled with data from file (containing traced species) and supplemented with values of the fix (CH2, H2, O2, N2) and of radical species (except from OH and HO2, as they are contained in the *nc file already).

REAL :: local_temperature(idim, jdim, kdim)
This array holds the local temperatures for every (i,j,k) grid cell.

3. Passing values from file to KPP internal routines

The function Get_Boxes(....) has now returned all information necessary for the full simulation domain given one point in time. Now, we can set up individual boxes within the full domain and pass respective concentration plus actual time and temperature values to KPP internal routines. The latter values are necessary, to allow for locally adapting rate constants by means of changing temperature and photolysis.

We choose an individual box and denote it by its indices (box_i, box_j, box_k). Respectively we get the local temperature on this box with

local_temperature = temperature_from_file(box_i, box_j, box_k)

and pass it via the function

CALL Initialize(local_temperature, act_hour),

which is a slight modification of the original function Initialize(), that sets global variables TEMP and TSTART to the actual temperature and time values.

Then we set initial values C (which shadows VAR and FIX) for box (box_i, box_j, box_k) by setting

C(1:82) = initial_conc_from_file(box_i, box_j, box_k, 1:82)
C = C*CFACTOR

The multiplication with CFACTOR is necessary, because KPP internally uses other unities than COSMO-ART.