# Python routines to convert L1 files in netCDF format

In the frame of the FRM4DOAS project, L1 radiance spectra have been submitted in netCDF format. The description of groups and variables is provided in the L1_format_20180613_v3.12.pdf document. BIRA-IASB provide Python routines to convert the files in ASCII format (used to submit the spectra measured during the CINDI-2 campaign) and the MFC BIRA binary format (daily files generated by MFC4QDOAS_Converter from original MFC binary or std format).

## *Function call*

```
python3 -c "from netCDF_L1 import *;  \
netCDF_L1_Convert(cfgFile, \
                  inpFile, \
                  ncPath, \
                  refFile='', \
                  slfFile='', \
                  tpFile='', \
                  cloudFile='', \
                  aodFile='', \
                  surfaceAlbedo=0., \
                  clbFile='', \
                  file_version='001', \
                  slfWavelength=0., \
                  file_format='ASCII', \
                  commentChar='#', \
                  solarAzimuthCorrection=0., \
                  viewingAzimuthCorrection=0.);"
```

with :

| | |
|---|---|
| cfgFile | ini file with some (key,value) to use as file attributes |
| inpFile | name of the input file (native format) |
| ncPath | name of the output path |
| refFile | optional, reference file for QDOAS calibration as keydata |
| slfFile | optional, slit function file provided by the data submitter as keydata |
| tpFile | optional, temperature/profile file |
| cloudFile | optional, cloud file (should include information on cloud coverage and heights) |
| aodFile | optional, aerosol data |
| surfaceAlbedo | optional, surface albedo |
| clbFile | optional, wavelength calibration file |
| file_version | optional, the file version on three digits |
| slfWavelength | optional, for 1D slit function, the wavelength at which the slit function has been measured |
| file_format | the native file format ('ASCII' or 'MFC-BIRA') |
| commentChar | optional, the character to use for comment in clbFile, refFile, slfFile |
| solarAzimuthCorrection | optional, the azimuth correction to apply to solar azimuth angle |
| viewingAzimuthCorrection | optional, the azimuth correction to apply to viewing azimuth angle |

Some comments about the parameters :

- Configuration file contains information that are used as netCDF file attributes but also to build the output file name (see example below)
- The format of "refile", "slfFile", "clbFile" is ASCII (it is the same format as the ones provided for CINDI-2)
- tpFile : is loaded by the __init__ function of "netCDF_L1_tp" class (could be modified according to the format of your file). The time vector has to cover the measurements times.
- cloudFile : is loaded by the __init__ function of "netCDF_L1_cloud" class (could be modified according to the format of your file). The file should contain at least, the time grid (fractional hour expected), the cloud coverage (in percent) and the cloud height (km)
- aodFile : is loaded by the __init__ function of "netCDF_L1_aod" class (could be modified according to the format of your file). The source code contains a function "read_aeronet" to retrieve information from files of the AERONET data base.
- surfaceAlbedo : only one value is expected
- commentChar : usually '#' or ';' depending on the user preference
- solarAzimuthCorrection, viewingAzimuthCorrection : profiling tools usually work with relative azimuth. A correction could be applied if solar and viewing azimuth are not given with the same convention or to suit requirements (for example, azimuth requested within 0..360 range).

newAzimuth=oldAzimuth+correction.

## *Example of config file*

[instrument]

instr_number=004
instr_channel=2
instr_type=maxdoas

[data_submitter]

institution=BIRA.IASB
pi_name=Michel Van Roozendael
pi_email=michel.vanroozendael@aeronomie.be
do_name=Caroline Fayt
do_email=caroline.fayt@aeronomie.be
ds_name=Francois Hendrick
ds_email=francois.hendrick@aeronomie.be

[location]

station_name=CABAUW
station_altitude_asl=70.
instr_latitude=51.97
instr_longitude=4.93
instr_altitude_asl=90.

## Example of call

```
python3 -c "from netCDF_L1 import *;   \
            netCDF_L1_Convert('./Example/cindi2_004_2_asc2netcdf.ini',
           './Example/bira_maxdoas_4_uv_spectra_cindi2_20160924_v1.asc',
           './Example','./Example/bira_maxdoas_4_uv_spectra_cindi2_20160924_v1.ref',
            slfFile='./Example/bira_maxdoas_4_uv_sltfct_cindi2_20160924_v1.asc',
            clbFile='./Example/bira_maxdoas_4_uv_spectra_cindi2_20160924_v1.ref',
            file_version='2',
            file_format='ASCII',
            commentChar='#',
            cloudFile='./Example/cloud.txt',
            tpFile='./Example/atm_std6.dat',
            aodFile='./Example/20180526_20180526_Brussels.lev15',
            slfWavelength=346.);"
```

## Conversion from other file formats

The netCDF_L1_Convert function (netCDF_L1.py) currently supports three formats for L1 spectra :

- ASCII : the ASCII format used to submit CINDI-2 spectra
- CCDEEV : the format used at BIRA-IASB
- MFC_BIRA : the format generated by the BIRA_IASB MFC4QDOAS_Converter tool

To add a new format, a function netCDF_L1from<new format> has to be created. This function should call your function that reads the spectra in the native format and make the equivalence between the data sets and the field of the class "netCDF_L1_records" that contains the description of the netCDF file.

Refer to netCDF_L1_LoadfromASCII or netCDF_L1_LoadfromMFC_BIRA to see how to proceed.

Then in netCDF_L1_Convert function, the following tests have to be completed with a call to netCDF_L1from<new format> :

```
  if (file_format=='CCDEEV'):
     L1_data=netCDF_L1_LoadfromCCDEEV(inpFile,clbFile);
  elif (file_format=='ASCII'):
     L1_data=netCDF_L1_LoadfromASCII(inpFile);
  elif (file_format==<new format>'):
     L1_data=netCDF_L1_Loadfrom<new format>(inpFile,<options>);
  elif (file_format=='MFC_BIRA'):
     L1_data=netCDF_L1_LoadfromMFC_BIRA(inpFile,clbFile,commentChar=commentChar);
  else:
     print('['+functionName+'] Unknown format '+file_format+' !!!');
```